★このページは紙芝居やCGアニメを解析するという新コーナーなのだ。プログラム的な解析は、マイクロキャビンのプログラマ中津さんに強引にお願いしちゃったぞ。

と、いうことで、いつのまにかこの教室をやることになっていた中津です。う〜む、世の中あなどれません。それでは、ともかく始めましょう。

今回はアニメ処理の基本中の基本ということで、裏VRAMとパレットの使い方を中心に解説します。解説は行番号ごとにやっていきますね。

10　画面などの初期化／SCREEN5(やはりアニメでコマ数が必要なら、裏VRAMが3画面ぶんとれるこのモードが最適。SCREEN7だと1枚しかとれませんからね)

20〜50　VRAMのページを切りかえながら読む(これで裏VRAMにもグラフィックが格納されたことになるわけです)

60　パレットを行50でロードしたデータで初期化しています。

70〜80、150　メッセージをグラフィック表示(グラフィックモードのときにはこうしないと文字が出せないので注意)

100、140　P＝0〜11のループ(もちろんFOR〜NEXTを使ってもOK)

110　(3<P<10)の間だけカラーコード10のパレット変更(背景がだんだん明るくなっていく効果です)

120　COPY命令で裏VRAMグラフィックを1コマだけ取り出して表VRAMに転送(座標や裏VRAMのページを計算だけでもとめている所にも注目です)

130　ループによるウェイト(タイミングをとるための待ち時間。ターボRだと高速モードでタイミングが狂うので注意してください)

といったところで今回はおしまい。ご意見、ご感想があったら「紙芝居&動画教室」あてにお手紙くださいね。　　(中津)

PROFILE

株式会社マイクロキャビン　企画開発課主任

中津泰彦

マイクロキャビンのプログラマーをやっています。MSX2版の『Xak』でデビューしました。以後『FRAY』、『幻影都市』と次々に作品を手掛けるプログラマー&ゲームデザイナーになってます。ときどきほかのMSX作品も手伝っていたりしていますよ。よろしく。

作品紹介

作品名/羽化

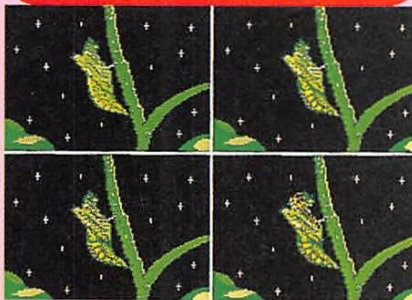
今回掲載した作品は、Mファン92年4月号で採用された『羽化』だ。サナギからチョウが誕生するシーンをみごとに再現し、最後でチョウになったとき、背景にアミがあらわれるというオチもある。

〈作者のコメント〉　この作品は、家のちかくにあったチョウのサナギを実際に見て作った作品なんです。　　(神奈川県・ためしい)

PROGRAM

```
10 DEFINTA-Z:COLOR15,0,0:SCREEN5
20 SETPAGE0,1:BLOAD"CG1.SC5",S
30 SETPAGE0,2:BLOAD"CG2.SC5",S
40 SETPAGE0,3:BLOAD"CG3.SC5",S
50 SETPAGE0,0:BLOAD"PLT.SC5",S
60 COLOR=RESTORE
70 OPEN"GRP:"FOROUTPUTAS#1
80 PSET(40,40):PRINT#1,"「うか」 ・アゲハ の いっし
ょうより"
90 PLAY"T60L107S0M30000G"
100 P=0
110 IFP>3ANDP<10THENCOLOR=(10,P-4,P-3,P-
2)
120 COPY((PMOD2)*128,((PMOD4)¥2)*106)-ST
EP(127,105),1+P¥4 TO (64,64),0
130 FORI=0TO3000:NEXT
140 P=P+1:IFP<12THEN110
150 PSET(180,180):PRINT#1,"おわり"
160 IFSTRIG(0)THENENDELSE160
```

裏VRAMをのぞいてみよう

それでは裏VRAMの中をのぞいてみよう。裏VRAMではページ(画面)が3つにわかれていて、その中に全アニメーションパターンが描かれた絵が入っていた。下の写真がそれ。写真のように1ページの中に4つのパターンが描いてあり、それを1カットずつ順番に表示させる。すると、羽化の瞬間が流れるというワケ。これだけ見ても流れがわかるよね。たいていの紙芝居作品は、VRAMの中にすべての絵を入れておき、表示中に合成したりする。ちなみにいつも見えている画面はページ0だ。

ページ1

ページ2

ページ3

『羽化』の裏ＶＲＡＭについての補足　裏ＶＲＡＭのページ１〜３には順に「ＣＧ１．ＳＣ５」〜「ＣＧ３．ＳＣ５」というグラフィックファイル（上のＢＡＳＩＣプログラムの行20〜40でロード）がロードされています。また、パレットデータの入っているファイルは行50に出てくる「ＰＬＴ．ＳＣ５」です。『羽化』はページ１〜３に書きこまれているグラフィックを必要なぶんだけページ０（表画面）にコピーすることで紙芝居を実現しています。

今月のPOINT
- ◆DATA文による省プログラム
- ◆背景と人物の合成テクニック

マイクロキャビン
中津泰彦

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。ＭＳＸ２版の「Ｘａｋ」でデビュー。以後「ＦＲＡＹ」、「幻影都市」など、次々を手掛けるプログラマ＆ゲームデザイナー。また、ときどきほかのＭＳＸ作品を手伝っているので、ＭＳＸにはなじみ深い人でもある。

作品紹介

拳舞（けんぶ）

Mファン92年8月号に採用されたこの作品は、シルエットであらわれる人物が、武道の舞をえんえんと続け、それに合わせて背景が変化していくものだ。今回この作品を付録ディスクに収録したので、じっさいに見てみよう。付録ディスクでは、CGコンテストのコーナーに入れてあるぞ。

〈作者のコメント〉 この作品は、人物と背景を同時に動かす実験として制作したものなんです。今は、デジタイズした映像をSCREEN5に変換し、アニメーションさせて、自分で撮ったビデオの編集などに使用して楽しんでいます。 （東京都・小澤考）

今月はまた過去の作品を取り上げた。いきなりむずかしいのをやってもしかたないしね。

早くも大好評の第２回（年末進行で今は12月あたま。だから本当のところは？ だったりするが）。

今回は「拳舞」である。この作品は、背景と人物の合成がシンプルなので、この教室にはうってつけなのだ。そのうえ、こいつぁDATA文の応用で実にコンパクト／長くややこしいプログラムになりがちなアニメ処理にぜひ使いたいテクニックだぁ／ ハァハァハァ、よし、始めるぞ／

10〜60 初期化、グラフィックのロード。ページは１〜３だ。

80〜130 グラフィックのデータをG、X、Yの配列にセット（※注欄外参照）。GXY（0〜8）が背景、GXY（9〜44）が人物などだ。

140〜160 GXY（6）を配列Pへ。全パレットを黒にして、配列Pを表示。これは、GXY（6）をバッファとして使う苦肉の策。バッファというのは、一時的にデータを保存／加工する場所のこと。まぁふつうはバッファを専用にあけておくこと。

180〜270 黒からパレットをフェードイン（DATAは行400〜430を使用）。最後に人物を合成。

290〜380 メインプログラム。行450〜790のDATA文から１データずつ読んでいる。以下はそのデータの意味。

（−１）なら再スタート。

（０〜８）背景データ（D）として登録。D＝6のときのみ背景は配列Pより。

（9〜44）背景Dに人物を合成表示。

（45）背景のみ表示。

（46〜）ウエイト。

これって、10000＋ループ回数としたほうがオレのシュミ。計算が楽なのだ。

ということで、かけ足になってしまったが、今回はこれにて。また来月に会おう／ （中津）

```
PROGRAM
10 DEFINT A-Z:COLOR 15,0,0
20 SCREEN 5:CLEAR 500
30 DIM G(45),X(45),Y(45),P(363)
40 FOR I=1 TO 3:SET PAGE 0,I
50 BLOAD "MAI-"+HEX$(I)+".GE5",S
60 NEXT I:SET PAGE 0,0
70 '
80 A=0:FOR J=0 TO 140 STEP 70:FOR I=0 TO
 170 STEP 85
90 G(A)=1:X(A)=I:Y(A)=J:A=A+1
100 NEXT I,J
110 FOR K=2 TO 3:FOR J=0 TO 140 STEP 70:
FOR I=0 TO 210 STEP 42
120 G(A)=K:X(A)=I:Y(A)=J:A=A+1
130 NEXT I,J,K
140 COPY(0,140)-(83,208),1 TO P
150 FOR J=0 TO 14:COLOR=(J,0,0,0):NEXT J
160 COPY P,0 TO (86,70),0
170 '
180 RESTORE 400
190 FOR I=1 TO 15:READ A(0),A(1),A(2)
200 FOR J=0 TO 7
210 FOR K=0 TO 2:IF A(K)>=J THEN B(K)=J
220 NEXT K
230 FOR K=0 TO 20:NEXT K
240 COLOR=(I,B(0),B(1),B(2))
250 NEXT J,I
260 FOR I=0 TO 400:NEXT I
270 COPY(0,70)-(40,138),3 TO (107,70),0,
TPSET
280 '----------- MAIN
290 RESTORE 450:FOR I=0 TO 700:NEXT I
300 READ C:IF C=-1 THEN 150
310 IF C>45 THEN FOR I=0 TO C-45:NEXT I:
GOTO 300
320 IF C<9 THEN D=C:GOTO 300
330 IF D=6 THEN COPY P,0 TO (0,140),1:GO
TO 350
340 COPY(X(D),Y(D))-(X(D)+83,Y(D)+68),G(
D) TO (0,140),1
350 IF C=45 THEN 370
360 COPY(X(C),Y(C))-(X(C)+40,Y(C)+68),G(
C) TO (21,140),1,TPSET
370 COPY(0,140)-(83,208),1 TO (86,70),0
380 GOTO 300
390 '----------- DATA
400 DATA 0,0,0 ,0,0,0 ,0,0,0 ,4,1,1
410 DATA 4,6,4 ,6,4,5 ,5,3,5 ,0,0,0
420 DATA 3,2,0 ,4,3,1 ,5,4,2, 6,5,3
430 DATA 7,6,4 ,0,0,0 ,7,7,7
440 '
450 DATA 8,11,10,9,80,10,11
460 DATA 35,36,37,38,150
470 DATA 37,36,35,11,10,200
480 DATA 9,10,11,12,13,14,15,16,17,18
490 DATA 19,20,28,27,12,11,10,9
500 DATA 10,11,12,13,14,36,37,38,80
510 DATA 37,36,35,11,10,200
520 DATA 7,9,80,10,11,12,27,34,150
530 DATA 6,39,300
540 DATA 3,16,15,14,13,2,12,1,11,0,10
550 DATA 33,150,6,41,300
560 DATA 7,10,11,12,27,28,29,30
570 DATA 31,32,150,6,40,300
580 DATA 0,9,80,10,11,12,13,14,15,16
590 DATA 17,18,19,20,21,1,22,2,23,3,24
600 DATA 4,25,5,26,260
610 DATA 7,27,28,29,30,31,32,28,29,30
620 DATA 31,32,28,29,30,31,32,28,27
630 DATA 12,11,10,33,80
640 DATA 4,45,3,45,2,45,1,45,0,45
650 DATA 15,16,17,18,19,20
660 DATA 8,14,13,12,11,10,9,300
670 DATA 6,34,120,7,38,130
680 DATA 6,33,120,7,31,250
690 DATA 8,15,16,17,18,19,20,28,27,12
700 DATA 11,10,9,250
710 DATA 7,9,10,11,35,36,37,38,37,36
720 DATA 35,36,37,38,37,36,35,11,12
730 DATA 27,28,29,30,31,32,28,20,19
740 DATA 18,17,16,15,14,13,12,11,10,9
750 DATA 200,10,11,12,13,14,15,16,17
760 DATA 18,19,20,19,18,17,16,15,14,13
770 DATA 12,11,10,9,600
780 DATA 6,42,400,43,44,300,43,42
790 DATA 4000,-1
```

裏VRAMをのぞいてみよう

写真から表示画面がページ０、背景がページ１、人物などがページ２、３にわかれているのがわかる。これは、背景と人物を別々にもってグラフィックを節約するのがねらい。だけど、その２つを合成する必要がでてくる。その方法がここに示したバッファ合成。これには、人物などのバックが透明色で描かれている必要がある。まずページ１の背景をページ１のバッファへ。次に人物をバッファに合成。最後にバッファをページ０に転送して終了。なぜ、こんな面倒なことをするのかというと、画面がチラつくのを防止するためなのだ。バッファを表示画面に移してみると実感としてわかると思うから実験してみてほしい。

ページ2
MAI−2.GE5

ページ3
MAI−3.GE5

ページ1
MAI−1.GE5

ページ0
表示画面

※注 配列Gはグラフィックのあるページ、配列X、Yは左上の座標を意味する。ここでは、ページG（n）の座標X（n）、Y（n）にあるグラフィックのことをGXY（n）と呼んでいる。プログラムの行340、360を参照。

★「紙芝居＆動画教室」への投稿募集中！応募要項は81ページ

いつもゆかいな中津先生がおくる、紙芝居解析コーナー！

紙芝居&動画教室

今月のPOINT

●動く人(物)の多重合成テクニック

マイクロキャビン
中津泰彦

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2版の『Xak』でデビュー。以後『FRAY』、『幻影都市』など、次々と手掛けたプログラマ&ゲームデザイナー。また、ときどきほかのMSX作品を手伝っているので、MSXにはなじみ深い人でもある。

今月は新しい投稿からピックアップした。けど、これがなかなかむずかしい。多重合成サンプルもあるので、じっくり理解してください

今月は投稿作品から解析。このぐらいの動画プログラムだと、長いので解析も大変である(マシン語まで入っていたのでさらに大変だった)。今後の投稿は私のことを考えて、プログラムにわかりやすくコメントを入れたりしてくれるとうれしいな(よろしく〜♪)。

さて、そろそろ解説に入ろう。ちなみに前回と文体がちがうが気にしないよーにっ/

10〜110　初期化である。画面をSCREEN5、マシン語ルーチンの登録(USRがパレット変更、USR2がVDPの直接操作)、グラフィックのXYデータ登録などをおこなっている。行90でやっているのは名前の変更機能であるがこれは割愛させていただく。

120〜140　グラフィックデータのロード、タイトルの表示(行890で実際に表示しています)。

150〜190　雨の絵をLINE文を使い、ランダムにタテ線を引いて作る。

200　雨の効果音(ザァーというやつですね)。雨の絵を作る途中にこの処理があるのは、演出の間を作るためと見ている。ちがう？

210〜250　色ちがいの雨の絵をLINE文を使いランダムにタテ線を引いて作る。

260〜340　男が歩く姿を合成して表示。行290は、パレットを雨の効果のために変更している。行300は、"オトコ2"をバッファへ。男を上下にゆらすためにPOKE文で転送元の座標を変更していることに注目(ここではマシン語でおこなわれるので、わかりづらいかも知れないが、その説明は行1060の解説を見て

SILENCE IN THE RAIN (by まいきー)

```
10 SCREEN5
20 CLEAR100,&HCFFF
30 DEFINTA-Z:OPEN"grp:"AS#1
40 DEFUSR=&HD000
50 DEFUSR2=&HD020
60 N$=" たけし"
70 IFSTICK(0)<>1THEN110
80 SCREEN1
90 INPUT"なまえをいれてくだ"さい。(4モシ")";N$
100 SCREEN5
110 COLOR15,0,0:CLS:PSET(90,90):PRINT#1,
"しはﾞし またれい":GOSUB980:GOSUB1060:GOSUB1310
:U=USR(AD)
120 SETPAGE0,1:BLOAD"チンモク0.SR5",S
130 SETPAGE0,2:BLOAD"チンモク1.SR5",S:GOSUB8
90
140 SETPAGE0,3:CLS
150 LINE(128,0)-(255,212),9,BF
160 FORI=0TO150
170 X=INT(RND(1)*127)*2:Y=RND(1)*212-20
180 LINE(X,Y)-(X,Y+RND(1)*4+30),11
190 NEXT
200 SOUND6,10:SOUND7,&B10110111:SOUND8,1
6:SOUND13,&B1101:SOUND12,100
210 FORI=0TO100
220 X=INT(RND(1)*127)*2:Y=RND(1)*212-20
230 LINE(X,Y)-(X,Y+RND(1)*4+30),10
240 NEXT:SETPAGE0,0
250 CLS
260 ' ●●●●●● オトコ アルク ●●●●●●
270 '
280 FORI=0TO60:II=60-I
290 U=USR(AD+30*(IMOD2))
300 POKE&HD340+18+16,(IMOD6)/1.8:U=USR2(
&HD340+16*2)
310 COPY(IMOD2,II*38MOD100)-(IMOD2+118,I
I*38MOD100+100),3TO(121,106),2,TPSET
320 U=USR2(&HD340+16*3)
330 NEXT
340 '
350 ' ●●●●●● オンナ フリムク ●●●●●●
360 J=0:SETPAGE0,2
370 U=USR2(&HD340+16*1)
380 FORI=0TO10:J=J+1:II=99-J
390 U=USR(AD+30*(JMOD2))
400 POKE&HD340+16*4+2,II*38MOD100:U=USR2
(&HD340+16*4)
410 COPY(0,106)-(82,209),1TO(122,106),2,
TPSET:GOSUB940
420 U=USR2(&HD340+16*5)
430 NEXT
440 '
450 J=J+1:II=99-J:U=USR(AD+30*(JMOD2))
460 POKE&HD340+16*4+2,II*38MOD100:U=USR2
(&HD340+16*4):COPY(86,106)-(168,209),1TO
(122,106),2,TPSET:GOSUB940:U=USR2(&HD340
+16*5)
470 '
480 FORI=0TO10:J=J+1:II=99-J
490 U=USR(AD+30*(JMOD2))
500 POKE&HD340+16*4+2,II*38MOD100:U=USR2
(&HD340+16*4)
510 IFI=1THENCOPY(121,98)-(127,104),1TO(
215,153),1
520 COPY(170,106)-(252,209),1TO(122,106)
,2,TPSET:GOSUB940
530 U=USR2(&HD340+16*5)
540 NEXT
550 IF STRIG(0) GOTO280ELSEJ=0:'GOTO440
560 '
570 '●●●●●●  オトコ ト オンナ ●●●●●●
580 '
590 U=USR2(&HD340+16*6)
600 U=USR2(&HD340+16*1)
610 FORI=0TO44:II=44-I
620 U=USR(AD+30*(IMOD2))
630 IF I=33 THENSOUND8,0:U=USR(AD+30*2):
TIME=0:FORJ=0TO1:J=-(TIME=200):NEXT:SOUN
D8,16
640 U=USR2(&HD340+16*7):J=I*2-(IMOD4=0)
650 POKE&HD340+16*8+4,J
660 U=USR2(&HD340+16*8)
670 COPY(80,0)-(85,74),2TO(J+75-I¥2,106)
,1
680 Y=(IMOD4)*2+(IMOD4=3)*5:COPY(167,Y)-
(254,Y+73),2TO( 90-J,106),1,TPSET
690 COPY(2+(II*2)MOD70 ,II*38MOD100)-(17
2+(II*2)MOD70,II*38MOD100+73),3 TO(0,106
),1,TPSET
700 U=USR2(&HD340+16*9)
710 NEXT
720 '
730 '●●●●●● オンナ シャベル ●●●●●●
740 SETPAGE0,0
750 U=USR2(&HD340+16*1):P=0:PP=0
760 FORI=0TO60:II=60-I
770 U=USR(AD+30*(IMOD2))
780 U=USR2(&HD340+16*0)
790 COPY(0,203)-(50,211),2TO(122,194),2
800 COPY(X(P),Y(P))-(X(P)+20,Y(P)+49),2T
O(130,154),2,TPSET
810 Y=II*38MOD100:COPY(2,Y)-(60,Y+103),3
TO(122,106),2,TPSET:COPY(70,Y)-(75,Y+90)
,3TO(122+123,106),2,TPSET
820 COPY(160,0)-(248,96),1TO(162,106),2,
TPSET
830 '
840 IF P=8 AND PP=0THENPP=1ELSEIF P=8 TH
ENPP=PP+1:IFPP=40THENPP=0:P=9
850 IF PP>0 THENPSET(110+PP*8+(PP>12)*12
0,150):I$=MID$(N$+"              チャックあいてるわよ。
                    ",PP,1):PRINT#1,I$:X=224-
INT(RND(1)*4)*11*(I$<>" "):COPY(X,86)-(X
+9,96),2TO(206,162),2ELSEP=P+IMOD2-((I-(
P>9))MOD2)*(P>9):IFP=14THENP=13
860 IFPP=10THENPSET(110,150):PRINT#1,"
   "ELSEIFPP=39THENPSET(85,150):PRINT#1,
"    THE END    ":SOUND7,135:SOUND9,16
870 U=USR2(&HD340+16*10)
880 NEXT:GOTO 760
890 U=USR(AD+30*3)
900 COPY(122,106)-(250,202),2TO(65,40),0
:TIME=0
910 FORI=5TO0STEP-1:COLOR=(6,3,I,6)
920 IFTIMEMOD(6-I)*10=0THENNEXTELSE920
930 '●
940 FORC=10TO11:FORK=0TO1:X=RND(1)*83+12
2:Y=RND(1)*14+106:PSET(X,Y),C:NEXTK,C:RE
TURN
950 '
960 '●●●●●● デ"ーター イッパ゚イ ●●●●●●
970 '
980 FOR I=0 TO 21:POKE &HD000+I,VAL("&H"
+MID$("DD214D012AF8F71600060f7E235ECD5F0
1142310F6C9",I*2+1,2)):NEXT
990 FOR J=0 TO3:FORI=0TO14:READ C$:AD=&H
D100+I*2+J*30:R=VAL(MID$(C$,1,1)):G=VAL(
MID$(C$,2,1)):B=VAL(MID$(C$,3,1)):POKE A
D,R*16+B:POKE AD+1,G:NEXTI,J
1000 AD=&HD100+30*0:RETURN
1010 '●●●●●●●●●●●● デ"ータ ●●●●●●●●●●●●
1020 DATA 000,112,765,654,544,433,446,33
4,223,456,677,567,723,651,666
1030 DATA 000,112,765,654,544,433,446,33
4,223,456,567,677,723,651,666
1040 DATA 000,122,566,455,344,233,344,23
3,122,455,455,566,344,344,566
1050 DATA 000,122,566,455,544,433,366,67
7,456,666,755,744,733,444,566
1060 FORI=0TO52:POKE &HD020+I,VAL("&H"+M
ID$("F30e993e02ed793e8fed790e99ed78e6012
0eeaf0e99ed793e8fed790e993e20ed793e91ed7
90c0c060f2af8f77eed792310fafbc9",I*2+1,2
)):NEXT
1070 RESTORE1090
1080 FORJ=0TO10:FORI=0TO14:READ A$:POKE
&HD340+I+J*16,VAL("&H"+A$):NEXT:NEXT:RET
URN
1090 DATA ,,,,7a,,6a,2
1100 DATA 85,,6A,,99,,c0  :'アオ1 -0
1110 DATA ,,,,28,,28,
1120 DATA D7,,6A,,,,c0    :'クロ0 -1
1130 DATA ,,,01,7a,,6a,02
1140 DATA 74,,65,,,,D0    :'オトコ -2
1150 DATA 7A,,6A,02,46,,28,
1160 DATA 74,,65,,,,D0    :'オトコP-3
1170 DATA A8,,,03,7A,,6A,02
1180 DATA 52,,67,,,,D0    :'オンナ -4
1190 DATA 7A,,6A,02,54,,28,
1200 DATA 52,,67,,,,D0    :'オンナP-5
1210 DATA ,,,03,80,,,03
1220 DATA 7e,,d4,,,,d0    :'アメC-6
1230 DATA 00,,00,,,,6a,01
1240 DATA aa,,49,,99,,c0  :'B&G -7
1250 DATA 57,,,02,06,,6a,01
1260 DATA 50,,49,,,,d0    :'B&G1-8
1270 DATA ,,6A,01,2A,,30,
1280 DATA AA,,49,,,,d0    :'B&GP-9
1290 DATA 7A,,6A,02,46,,28,
1300 DATA 80,,60,,,,D0    :'サイコ"-P
1310 DIM X(13),Y(13):RESTORE1310:FORI=0T
O13:READX(I),Y(I):NEXT:RETURN
1320 DATA 0,1,0,51,0,101,0,151,22,1,22,5
1,22,101,22,151,44,1,44,51,44,101,44,151
,66,101,66,151
```

※『SILENCE IN THE RAIN』は付録ディスクに収録しています。⇨112ページ

ほしい)。行310は、その上に先ほど作った“雨の絵”を重ね、そのとき、雨を動かすために転送元の座標を少しずつ変更しているのだ。行320は更に合成された絵をバッファからマシン語で表示スクリーンへ転送。ようするに転送元や転送先の座標をうまくいじってやると動いて見えるわけなのである。わかりづらければ、サンプルプログラムを見てね。

350〜430　たたずむ女の子。しばらくの間ループしている。

450〜560　女の子が振り向く。行550でヘンなことをしているが、スペースキーを押しているとわかると思う。

570〜720　男と女の子がすれちがう。I＝33の時だけ時間を止めて白黒にしている効果がにくいぞ、こらこら。

730〜880　去って行く男の後ろ姿に女の子がひと言。オチがついてオシマイ。ここではループのカウンタIの変化に合わせていろんな事をさせている。ただし、あんまりきれいなプログラムじゃないので修行するように！

890〜950　タイトルを表示するサブルーチン。

960〜1050　パレットを変更するマシン語をアドレス&HD000に登録している。パレットのデータは、行1020からのデータを&HD100から同じく登録している。このマシン語を使うときは、パラメータにパレットデータのアドレスを入れてコールすること。

1060〜1300　VDPを制御するマシン語をアドレス&HD020に登録している。行1090からが、VDPの制御データである。0〜10までの制御データが&HD340から登録される。このマシン語を使うときは、パラメータにVDP制御データのアドレスを入れてコールすること(VDPの制御データはR#32からのものです)。

1030〜1320　アニメ用の座標データ。　と以上である。(中津)

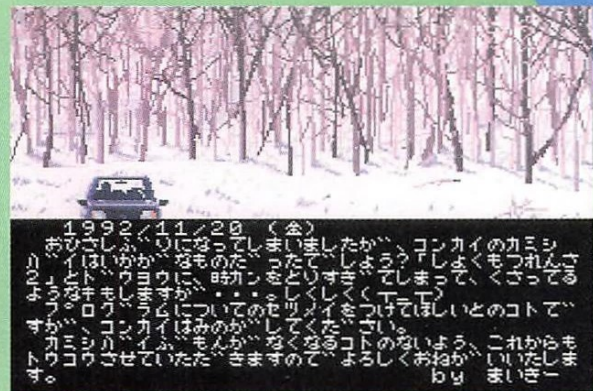
作品紹介

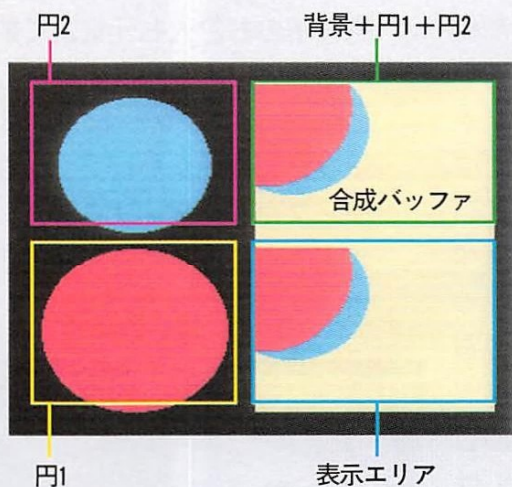
SILENCE IN THE RAIN

今月採用した作品は、CGコンテスト時代からの紙芝居の雄といわれていた、まいきーの作品。久しぶりの彼の作品は、『SILENCE IN THE RAIN』(雨の中の沈黙)だ。雨の中、カサをわすれたのか、ぬれながら歩く男の子。それをみかける女の子。2人はなんとなく好きあっていそうな雰囲気であることがうかがえる。そんななか、2人ともすれちがってしまう。そして女の子が一言……「チャックあいてるわよ……」。雨の音が強くなる。男の子はガクッとうずくまってしまった。という笑える紙芝居なのだ。

作者のコメント

今回の作品は雨を表現するために、アニメーションの高速化をこころみました。そのためのマシン語のVRAM転送部分はLINKS会員の方から助言をいただきました。ありがとうございます。(^_^)。あいかわらずのオチですが、ゆるしてください。これからもときどき顔を出すと思いますんでよろしく。

広島県・まいきー

↑投稿ディスクの中に編集部あてのお手紙が入っていた。グラフィック付きの豪華なお手紙だ。なんかうれしい

○サンプルプログラムで、多重合成を学んでみよう！

サンプルプログラム

```
5 DEFINT A-Z
10 SCREEN 5:SET PAGE 0,0:COLOR 15,0,0
20 CIRCLE(64,50),40,7:PAINT(64,50),7
30 CIRCLE(64,150),49,9:PAINT(64,150),9
40 F=0
50 FOR I=0 TO 50
60 X=I*2:Y=I:X2=X+20:Y2=Y+40
70 LINE (128,0)-(255,99),11,BF
80 IF F THEN GOSUB 130:GOSUB 150 ELSE GO
SUB 150:GOSUB 130
90 COPY (128,0)-(255,99) TO (128,100)
100 NEXT
110 F=F XOR 1
120 GOTO 50
130 COPY (126-X2,198-Y2)-STEP(X2,Y2) TO
(128,0),,TPSET
140 RETURN
150 COPY (X,Y)-STEP(127-X,99-Y) TO (128,
0),,TPSET
160 RETURN
170 'SAVE"sample.bas
```

さて、チンプンカンプンだった人も、ちょこっと理解した人も、このサンプルプログラムを見てほしい。このプログラムは、動くものを合成するときのサンプルです。

10〜30　円を描いて下準備をしています。

50〜120　円がすれちがいます。1回目と2回目で合成のされ方がちがうので、よく見てね。

130〜140　円1転写サブ。

150〜160　円2転写サブ。

複雑なことをしているように見えますが実にかんたんなプログラムです。ですから、いろいろ改良して合成の手法を実際に身につけてください。

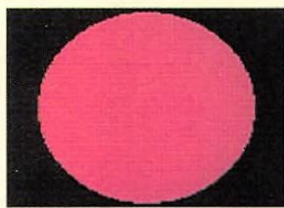
円だけの絵でもいろいろ面白いアニメーションは作れると思います。絵心のない方も、アニメーション技術でこのコーナーに投稿してみてくださいね。お待ちしてます。

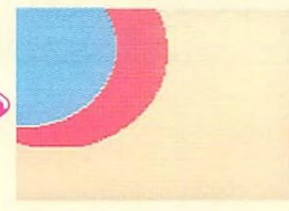
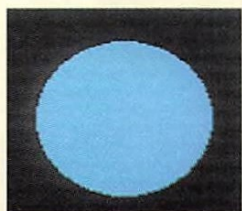
合成の手順

1.まずは合成バッファを背景色でぬる

ここでは、このサンプルプログラムを使って合成の手順を解説します。先月号で出てきたバッファを、なぜわざわざ用意してバッファで合成するのか？　といったことがこのサンプルでわかっていただけると思います。

まずは、合成バッファに残った前の絵を消すために背景色で塗りつぶしています。これをやめると残像が残るようになります。

2.円1の一部を取り出し合成バッファに合わせる

まずは右の写真のように、背景だけとなったバッファの上に、円Iの一部を転送してきて重ね合わせています。これはわかりますよね。

3.円2の一部を取り出し合成バッファに合わせる

背景と円Iの重なったさらにその上に、円2の一部を転送して重ね合わせて終了です。これを表示エリアに転送している。

★「紙芝居&動画教室」への投稿募集中！応募要項は81ページ

紙芝居&動画教室

今月のPOINT
お絵かきさんから動きに進級

今月から作品解析はやめにして、紙芝居の作り方を1から指導します。いままでCGしか描いたことのない人で紙芝居に興味がある人は、この連載講座を読んでいけば、最後は紙芝居を作れるようになるぞ！

➜P108の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2版の『Xak』でデビュー。以後『FRAY』、『幻影都市』など、次々と手掛けたプログラマ&ゲームデザイナー。また、ときどきほかのMSX作品を手伝っているので、MSXにはなじみ深い人でもある。

プログラムがRAMにいくなら画像データはVRAMへ

パッと見てくれるとわかると思うが、今回より内容一新／初心者にもわかる誌面作りをめざすことになった。気分も新たにがんばるぞ！　の中津です。

で、なぜ内容一新かというと、そもそも紙芝居を楽しむ人をふやす目的で始めたこのコーナー、どうやらムズかしく突っ走りすぎたみたいで、「わかんなーい！」のおハガキが続出してしまったのだ。これでは意味がないというわけで、路線変更とあいなったわけである。

かんたんにいえば、いままでのような投稿のプログラム解析はしない。投稿作品は楽しんでもらうを前提に紹介だけにした。そしてここでは、いままでCGしか描いたことのない人が紙芝居を作れるようにと、わたくし中津がお教えしていきたい。「中津先生〜♡」なんてよばれたら、ゾクゾクしちゃうぞ。わかんないことがあったら、どんどん意見をよせてちょうだい。

それでは、新装開店第1回は「RAMとVRAM」から始めてみたいと思う。

RAMっていうのは知ってるよね。キミたちがプログラムとか打ったらコンピュータがそれを記憶しておく電子部品のことだ。RAMは、いつのまにか一般用語に化けてしまって、今ではデータをたくわえる場所とか入れ物の名前という事になってしまっているけどね。で、データといっても、グラフィックや数値、プログラムなども含むわけで、まあ、何でもありありの世界ではあるのである。

しかし、わがMSXでは、RAM64K、VRAM128Kとかいって、RAMのほかにVRAMというヤツがいる。

これがどういうヤツかというと、RAMにはプログラムや数値が記憶されるが、キミたちが描いたグラフィックとかは、VRAMに記憶される約束になっているのだ。VRAMとは、VIDEO・RAMのこと。つまり画像用のRAMという意味である。

キミたちのMSXのモニタに表示されてるグラフィックはすべてVRAM上にあって、それを動かすプログラムがRAM上にある。これは、BASICであろうと、市販のゲームであろうと変わらない基本中の基本であるといえるだろう。

とにかくVRAMとは、絵や画像＝グラフィックを入れるところであると覚えてもらいたい。そしてさらに、VRAMの実体にせまってみよう。

MSXにはRAMとVRAMがある

```
10 SCREEN 0
20 PRINT"NAKATSU"
30 GOTO 20
```

MSXでは、プログラムはRAMに記憶されるが、画像データは画像データ専用のRAM（VRAM）に記憶されるしくみになっているのだ

AFTER SCHOOL
紙芝居倶楽部 今月の投稿作品より

ここのコーナーはみんなから来た投稿作品を紹介していく部分です。さて今月は小沢考クンの投稿で「北風と太陽」です。これはイソップ童話「北風と太陽」のパロディなので、原作を知らない人は、ぜひそちらを読んでからお楽しみください。読むと10倍は楽しめますよ。

作者より一言　東京都／小澤考(22歳)

この作品は、オンメモリで動かすために画像データを3画面につめこむのに苦労しました。メッセージは漢字表示させたかったのですが、裏VRAMに入りきれなかったので断念しました。作中の登場人物は、すべて自分の顔を取りこみ、SCREEN8からSCREEN5に変換、頬や口を修正し、キャラクタの個性を出したものです。また、眉をつりあげたり口もとをニヤけさせたりして、表情をおおげさにしています。ちなみに本人はこんな顔ではありません。たぶん……。

【北風と太陽】

SCREEN5／DD倶楽部／画面4分の1使用

①

北風：どちらが旅人のコートをぬがせられるか勝負しないか？

②

太陽：ようし、いいとも！

⑦

太陽：なんてガマン強い男なのだろう。
北風：あのコートをぬがすのは、とても無理だな。

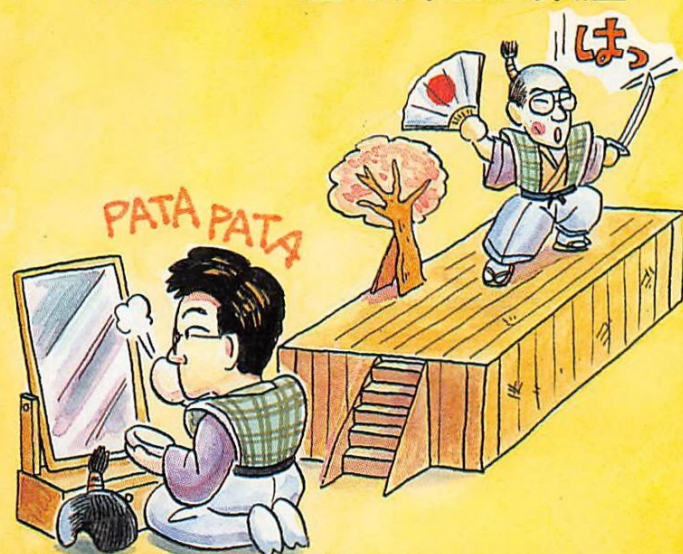
MSXのVRAMには舞台と楽屋がある

VRAMには、1画面分のグラフィックだけではなく、何画面分かのグラフィックを記憶しておくことができる。キミたちがよくゲームとかしているとき、目の前で表示されている画像のほかに、すでにちがう画像がVRAMには記憶されているのだ。

これを、見えている部分を舞台、見えない部分を楽屋と考えてもらう。つまり、キミたちの目の前に表示されている画像が舞台であり、裏でかくれている画像が楽屋ということだ。

これを紙芝居でいうと、表示して動いている部分が舞台で、アニメパターンがしまってあるのが楽屋にあたる。この教室でも最初に「裏VRAMをのぞいてみよう」ってやったよね(わかんない人は1月号か2月号を見よう)。あれが楽屋なんだ。

BASICの命令でCOPYというのがある。これにはいろいろな機能があるのだが、そのひとつに、楽屋から舞台にグラフィックを移す機能がある。

これを使って、何枚かのグラフィックを次々と表舞台に移すことが、見ている側からは紙芝居や動画になるというわけなのだ。ねっ、かんたんでしょ？

VRAMには、画面に表示している部分が舞台とすると、裏で次にだす画面を準備できる楽屋ももっている。紙芝居は実際楽屋にアニメパターンを描きためておいて再現するのだ

紙芝居するならSCREEN5がベスト

さて、紙芝居をする場合問題となるのはコマ数である。VRAMには何画面かのグラフィックを記憶できると書いたが、やっぱり限界があるのでたくさんのコマをわりふれない。

キミたちがよくCGを描くときに使うSCREEN7のモードでは、じつは2画面分しかVRAMは記憶してくれない。しかもその1画面分は表示画面として使ってしまうので、楽屋に入るアニメパターンは残りの1画面分しか置いておけないのだ。

これが、SCREEN5となると、なんと4画面分も記憶できるぞ。その中の1画面が表示画面として使われても、のこり3画面分のグラフィックを楽屋に待機させておけるので、この量はあなどれない。ドットがアラいから……なんていってられない。現にいままでの紙芝居投稿者は、ほとんどがSCREEN5を使っている実例がある。紙芝居をするなら、ぜひSCREEN5を使ってもらいたい。

次回の授業は、実際に動画を用意して、BASICでアニメさせる基本的な方法について、教室を開こうと思います。では次号！　(中津)

ドットはアラいがパターンは多い!

ふだん、みんながCGとして描くSCREEN7は全部で2画面分の絵しか記憶できない。しかし、SCREEN5だと4画面分も記憶できるのだ。紙芝居をするならSCREEN5がベストなのだ！

SCREEN5の場合

SCREEN7の場合

質問、投稿どんどんください

新装開店して第1回目の授業、いかがでしたか？　きっと紙芝居をサクサク作れる人にはバカバカしいページかもしれませんがCGしか描けなくてそれを動かしてみたいという人は多いのです。これからも今回みたいに教えていきたいと思いますので、「こんなことやってほしい」とかありましたら編集部あてにおたよりください。投稿もお待ちしています。　(中津)

★「紙芝居&動画教室」への投稿募集中！応募要項は81ページ

紙芝居&動画教室

新教室になっての2時限目は、プログラミング方法の基本を教えるよ。今回の授業で出てくるBASIC命令は大きく2つ。この2つさえ覚えてしまえば、それだけでアニメさせることができちゃうんだ。

→P108の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

講師プロフィール
株式会社マイクロキャビン・企画開発課主任。MSX2版の「Xak」でデビュー。以後「FRAY」、「幻影都市」など、次々と手掛けたプログラマ&ゲームデザイナー。また、ときどきほかのMSX作品を手伝っているので、MSXにはなじみ深い人でもある。

キミはここの楽屋で待機せよ(SET PAGE)

こんにちは、中津です。今月は、BASICでじっさいどのように紙芝居させるかを中心にお教えします。

そして、なんと！ キャビンオリジナルのグラフィック付きでサンプルプログラムをつけてみました。と～ぜん、グラフィッカーはフレイ＝末永大先生なのだぁ。大感謝！

と、いうことで、さっそく授業に入りましょう。

まず、BASICでプログラムする場合は、SCREENモードの設定からはじめます。書式は、

SCREEN スクリーンモード番号

書き方は、SCREEN5といった感じにそのまま書きます。

この命令は、SCREENの後ろにこれから紙芝居で使用するSCREENモードを番号で設定するのです。まぁ紙芝居だからSCREEN5がベストでしょう(前号参照)。

次にSETPAGEという命令を使います。この命令は、画面ページの切りかえをするものです。例えば、前号の授業で、SCREEN5は表示画面1つと絵をしまっておく楽屋3つで、全部で4ページの絵をVRAMは記憶してくれるといったよね。この4つのページは、そのまま数字で0，1，2，3と部屋番号みたいなものがついているんだ。そこで書式をかくよ。

SET PAGE 表示ページ，書きこみページ

書き方は、SET PAGE 0，1のように書きます。

んじゃ説明。表示ページとは、じっさい画面に表示されるページのこと。ここに0と入れれば0ページが画面に表示され、2と入れれば2ページが表示される。一般に0ページを表示用にして、あとのページは楽屋にする人が多いかな。

書きこみページとは、これから読みこむグラフィックはこのページに入って待機しなさいという意味と思ってね。本当はこのあとBLOADでグラフィックデータを読みこませるわけ。1と書いておけば読みこむグラフィックは1ページ目に入り、2と書けば2ページ目に入る約束になっているよ。

例

表示ページ，書きこみページ

SET PAGE0,1

訳:ページ0を表示中に、ページ1を描く

SETPAGEは、ページの切りかえの命令です。SETPAGEの後にいれる数字は、表示ページ，書きこみページを数字で入れます。上の例文ですと、画面にページ0を表示しておいて、その間にページ1を描きこむという指定をしていることになります。

AFTER SCHOOL

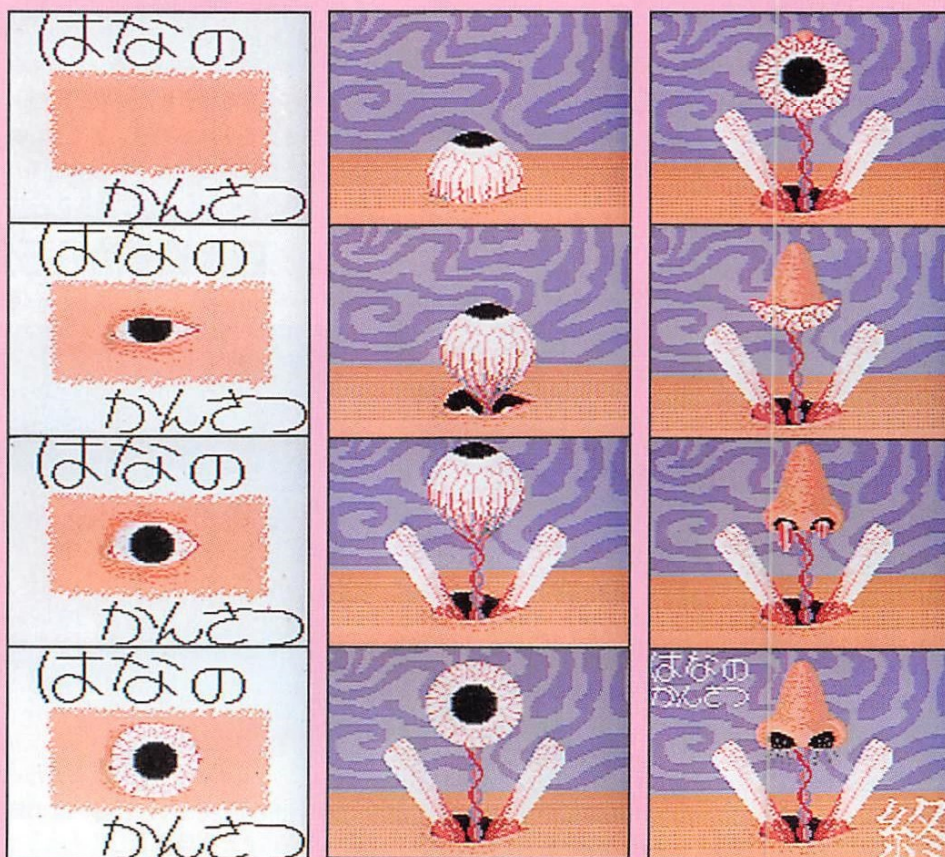
紙芝居倶楽部～今月の投稿作品より～

最近、ちょっと投稿がふえてきたのでうれしい中津です。ふえたといってもちょっとだけですよ、ちょっとだけ。これからもドンドンふえるように、みんなの投稿をお待ちしています。

さて今月の投稿作品からは、FS-A1TF作『はなのかんさつ』を選びました。僕ははじめ、花の開花の瞬間を予想していたのですが、花が鼻だったのには裏切られ、思わず力がぬけちゃいました。使用したツールはSIIANIMEだそうです。SCREEN5ですね。プログラムを見るとわかりますが、この作品はオールBASICで、今日の授業でやったCOPY～を使ったやり方ですので、参考にしてみてくださいね。

作者より一言

初めての投稿で、どうなることかと思いましたが採用されてとてもうれしいです。さてこのプログラムは単純ルーチンなので自分でも気に入ってます。それにスプライトを使い、ページ0にもいろいろ見えない所で使っています。ヒマな人は読みこんで見てください。動きはできるだけなめらかにと作りましたが、後半は少しザツです。最後に、太田君、伊藤君、元気ですか？ 福井でもがんばるよー。
福井県／FS-AITF(14歳)

【はなのかんさつ】
SCREEN5・4分の1画面サイズ
福井県・FS-A1TF(14歳)

ささ、出番ですよぉ(COPY〜)

ここでは紙芝居によく使うBASIC命令の解説をしているので、ちょっとムズかしいです。じっさいどのようにプログラムに入れるのかは、下のサンプルのカコミを見てください。では、次の解説に入りましょう。

アニメはちょっとずつちがった絵を連続で表示させて動きをみせるもの。ちょっとずつちがう絵は描けるけど、どうやって表示させたらいいのかわからない人が多いんじゃないかな？

ズバリ、表示するときは楽屋(裏ページ)から舞台(表示画面)へグラフィックを持ってくればいいのです。これを行うBASIC命令がCOPY命令。この書き方は、

COPY 転送元座標，転送元ページ TO 転送先座標，転送先ページ

という書き方をします(右の例も見てね)。

この命令は、表示画面に持ってくる絵をすべて座標で指定しています。座標はグラフィックツールのスモールルーペとかにでてくる座標とまったくいっしょ。そのまんまです。

さてそのなかで注意をひとつ。転送元座標の指定で、絵のデータの左上と右下の座標を指定する必要があります。持ってくる絵の大きさを指定しないといけませんからね。

下の例を見てください。まず転送元座標のカッコで囲まれた部分ですがこれは(左上のX，左上のY)ー(右下のX，右下のY)という意味です。

転送先座標は表示画面の表示位置指定です。画面のまん中あたりに表示するのがいいでしょう。座標の指定がしてあるカッコがひとつしかありませんが、これは左上の座標だけでいいのです。だって転送元の指定で、絵の大きさはすでに決まっていますからね。

あっ、ちなみに転送する絵はすべて四角で囲まれた絵です。丸くくりぬいて表示するなんて特殊なことはできませんよ。まぁ、例の訳を見たほうが早くわかるかな？

例

転送元座標　転送元ページ　転送先座標　転送先ページ

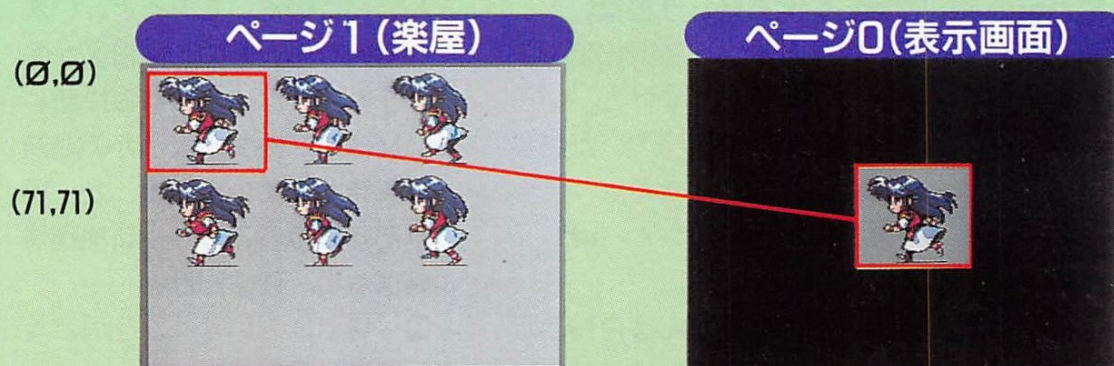
COPY(Ø,Ø)-(71,71),1TO(97,7Ø),Ø

訳：ページ1の中にある、座標Ø、Øから71、71までの四角く囲った部分を、ページ0上の座標97、7Øに表示しなさい

ページ1(楽屋)

(Ø,Ø)

(71,71)

ページ0(表示画面)

COPY〜はいろいろな使われ方がありますが、この場合はグラフィックの転送をしている命令です。例文と訳の座標をよくてらし合わせて見てください。これを座標をかえていちいち指定すれば、アニメができるわけです。

フレイのサンプルで、実際の組み方を学ぼう

ではサンプルプログラムで、プログラムの書き方を見てみましょう。

10　SCREENを5に設定。

20　ページ0を表示して、ページ0をCLSします。これは、画面をクリアするもので、書きこみページをクリアすることができます。

30　画面はページ0を表示。行40で読むグラフィックデータをページ1に入れろと設定しています。

40　グラフィックデータ読みこみ。読みこんだグラフィックは行30で指定した書きこみページ(ページ1)に転送されるので、まだ見えないはずですね。

50　読みこんだグラフィックのカラーパレットをセット。

60　さて、ここからページ1に入っている絵をページ0(表示画面)に転送。座標の指定は上の例を参照してね。これは、6つあるフレイのアニメパターンの一番左上の絵を表示させています。

70　ここはウエイト(待ち時間)のコントロールに使っています。これを入れないと、表示した次の瞬間には次の絵が出てしまうので速すぎて何が何だかわからなくなってしまいます。ちなみに50という数字を大きくするほど待ち時間が長くなります。

80〜170　転送作業を1パターンずついちいちやっています。

180　行60にもどって永遠にフレイを走らせているわけです。

SAMPLE.BAS

```
10 SCREEN 5:COLOR 15,1,1
20 SET PAGE 0,0:CLS
30 SET PAGE 0,1
40 BLOAD"f1.ge5",S
50 COLOR=RESTORE
60 COPY (0,0)-(71,71),1 TO (97,70),0
70 FOR I=0 TO 50:NEXT
80 COPY (72,0)-(143,71),1 TO (97,70),0
90 FOR I=0 TO 50:NEXT
100 COPY (144,0)-(215,71),1 TO (97,70),0
110 FOR I=0 TO 50:NEXT
120 COPY (0,72)-(71,143),1 TO (97,70),0
130 FOR I=0 TO 50:NEXT
140 COPY (72,72)-(143,143),1 TO (97,70),
0
150 FOR I=0 TO 50:NEXT
160 COPY (144,72)-(215,143),1 TO (97,70)
,0
170 FOR I=0 TO 50:NEXT
180 GOTO 60
```

フレイがひたすらペタペタ走ります。次回はここから転ばせてみようかな？

質問、投稿どんどんください

いかがでした？　SETPAGEとCOPY〜さえ覚えてしまえば、かんたんなアニメーションなら作れるようなります。けっこう座標指定がめんどうくさいかもしれませんが、これがMSXでアニメーションさせるいちばん楽な方法です。また、今回のサンプルプログラムはひじょうに短く、もっともかんたんな例です。たとえばフレイの絵の部分を自分で描きかえて、このプログラムを走らせてもいいわけです。いろいろやってみて理解を深めてください。

みなさん、ぜひこのプログラムを参考に、アニメに挑戦してみてほしい。キミの学習結果を待ってるぞ！　じゃ、まったねー。　(中津)

★「紙芝居&動画教室」への投稿募集中！応募要項は81ページ

紙芝居&動画教室

今回のPOINT
プログラムをきれいにまとめてみよう

3回目にしてもう7月。この教室も1学期をおわらせてひとくぎり。1学期最後の授業は、前回の応用でプログラムをいかにみじかくまとめるかだ。じっくり読んで、まとめかたのコツを覚えよう!

➜P108の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

ページ2、ページ3も使ってアニメパターンをふやそう

こんにちは、中津です。夏休みももう目の前。楽しい時間がいっぱいとれる夏休みは、ぜひ紙芝居作りにはげんでみてくださいね。

それでは、授業に入りましょう。今回はおもに前回やったことの応用です。前回は1画面のCGデータを転送してアニメを実現しましたが、今回は3画面のCGデータをあつかいます。

といっても、転送作業をおこなうCOPY〜の文法に、その画面のページを指定してやればいいだけなのです。ここでわからなくなっちゃた人は、前回にもどってもういちどCOPY〜の使い方を学習しましょう。

今回のメイン学習は、プログラムのじょうずなまとめ方を教えます。1画面アニメーションならプログラムもけっこうみじかくてすみますが、3画面ともなると、どうしても大きくなりがちです。

右のプログラムを見てください。これは、前回の走るフレイちゃんに続いて作ったプログラムで、3画面分のCGを使ってアニメーションをします。プログラムが長いのがわかりますよね。ではこれを、いかにしてみじかくするのでしょうか?

⬆今回のサンプルプログラムはフレイが走って転びます。CGはもち末永先生だぞ

```
10 SCREEN 5:COLOR 15,1,1
20 SET PAGE 0,0:CLS
30 SET PAGE 0,1
40 BLOAD "f1.ge5",S
50 SET PAGE 0,2
60 BLOAD "f2.ge5",S
70 SET PAGE 0,3
80 BLOAD "f3.ge5",S
90 COLOR=RESTORE
100 SET PAGE 0,0
110 COPY (0,0)-(71,71),1 TO (97,70),0
120 FOR I=0 TO 50:NEXT
130 COPY (72,0)-(143,71),1 TO (97,70),0
140 FOR I=0 TO 50:NEXT
150 COPY (144,0)-(215,71),1 TO (97,70),0
160 FOR I=0 TO 50:NEXT
170 COPY (0,72)-(71,143),1 TO (97,70),0
180 FOR I=0 TO 50:NEXT
190 COPY (72,72)-(143,143),1 TO (97,70),0
200 FOR I=0 TO 50:NEXT
210 COPY (144,72)-(215,143),1 TO (97,70),0
220 FOR I=0 TO 50:NEXT
230 COPY (0,0)-(71,71),2 TO (97,70),0
240 FOR I=0 TO 50:NEXT
250 COPY (72,0)-(143,71),2 TO (97,70),0
260 FOR I=0 TO 50:NEXT
270 COPY (144,0)-(215,71),2 TO (97,70),0
280 FOR I=0 TO 50:NEXT
290 COPY (0,72)-(71,143),2 TO (97,70),0
300 FOR I=0 TO 50:NEXT
310 COPY (72,72)-(143,143),2 TO (97,70),0
320 FOR I=0 TO 50:NEXT
330 COPY (144,72)-(215,143),2 TO (97,70),0
340 FOR I=0 TO 50:NEXT
350 COPY (0,0)-(71,71),3 TO (97,70),0
360 FOR I=0 TO 50:NEXT
370 COPY (72,0)-(143,71),3 TO (97,70),0
380 FOR I=0 TO 50:NEXT
390 COPY (144,0)-(215,71),3 TO (97,70),0
400 FOR I=0 TO 50:NEXT
410 COPY (0,72)-(71,143),3 TO (97,70),0
420 FOR I=0 TO 50:NEXT
430 COPY (72,72)-(143,143),3 TO (97,70),0
440 FOR I=0 TO 50:NEXT
450 COPY (144,72)-(215,143),3 TO (97,70),0
460 FOR I=0 TO 50:NEXT
470 FOR I=0 TO 1000:NEXT
480 GOTO 110
```

このプログラムは前回の続きで、3画面分のCGを楽屋に読みこんでからアニメを実現しています。前回から追加したプログラムは行230以降で、それぞれ下に示したとおり、ページ2、ページ3の転送作業をいちいちおこなっています。

ページ1

ページ2

ページ3

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2版の『Xak』でデビュー。以後『FRAY』、『幻影都市』など、次々と手掛けたプログラマ&ゲームデザイナー。また、ときどきほかのMSX作品を手伝っているので、MSXにはなじみ深い人でもある。

AFTER SCHOOL

紙芝居倶楽部〜今月の投稿作品より〜

【餅つき】

東京都・小澤考(22歳)
SCREEN5・4分の1画面サイズ

作者より一言

4−5月号の「北風と太陽」に続いて、また取りこみ画像をSCREEN5で修正した作品です。今回は音楽や効果音のないさびしさをカバーするために、静止画中でつねに文字を動かすことで、作品全体の流れが止まらないようにしています。これは某テレビ局の「ウゴウゴルーガ」という番組を参考にしました。この作品は、誌面で見るよりも先にプログラムを走らせてみることをオススメします。そのほうがインパクトが強くて楽しめますよ。

東京都/小澤考(22歳)

今回の投稿作品からは、常連の小澤考クンが選ばれました。取りこみ画像を使用した作品で、考クンお得意ですね。この登場人物は作者なのでしょうか? 前に採用された「北風と太陽」は、お話つきの紙芝居でしたが、今回はお話はありません。アニメーションのみでギャクをやっています。見せ方がじょうずで感心しました。あとこの作品は、文字をふよふよ動かしているところがあります。ぜひ注目しましょう。

今回は考クンの作品のほかに、なかなかおもしろい作品がいくつか来ました。まず兵庫県のCOMEON太郎クンの作品で『忍』。コマ自体は小さいアニ

プログラムをみじかくまとめるコツ

いきなりですが、右のプログラムを見てください。じつはこれ、左ページのプログラムとまったく同じ働きをします。このまとまりかたのヒミツはなんでしょう？

まずこのサンプルプログラムの場合、ページ1、ページ2、ページ3にあるアニメパーツの転送元座標がすべて共通という点があげられます。具体的にいいましょう。左ページのプログラムの行110と、行230と行350は、転送元ページの指定がそれぞれ1、2、3となっているだけで、座標の指定はすべていっしょです。

アニメパーツは1ページで6パターン分あります。この6パターンの転送命令だけをプログラムに組んでおき、ループするようにして、プログラムが1周目ならページ1から転送、2周目ならページ2から転送といったように、コンピュータに自動的にカウントさせれば、みじかくまとめることができます。

さて、これをおこなうには条件分岐の命令を使います。それは、ＩＦ～ＴＨＥＮ命令とそれ対応のハコを用意するのです（くわしくは右の解説で）。かんたんにいうと、ハコの中身が1ならばページ1から転送、2ならページ2から転送ということで、それをチェックする命令が、ＩＦ～ＴＨＥＮ命令ということです。

ここまでまとまるサンプルプログラム

SAMPLE2.BAS

```
10 SCREEN 5:COLOR 15,1,1
20 SET PAGE 0,0:CLS
30 SET PAGE 0,1
40 BLOAD "f1.ge5",S
50 SET PAGE 0,2
60 BLOAD "f2.ge5",S
70 SET PAGE 0,3
80 BLOAD "f3.ge5",S
90 COLOR=RESTORE
100 SET PAGE 0,0
110 J=1
120 COPY (0,0)-(71,71),J TO (97,70),0
130 FOR I=0 TO 50:NEXT
140 COPY (72,0)-(143,71),J TO (97,70),0
150 FOR I=0 TO 50:NEXT
160 COPY (144,0)-(215,71),J TO (97,70),0
170 FOR I=0 TO 50:NEXT
180 COPY (0,72)-(71,143),J TO (97,70),0
190 FOR I=0 TO 50:NEXT
200 COPY (72,72)-(143,143),J TO (97,70),
0
210 FOR I=0 TO 50:NEXT
220 COPY (144,72)-(215,143),J TO (97,70)
,0
230 FOR I=0 TO 50:NEXT
240 J=J+1
250 IF J<4 THEN GOTO 120
260 FOR I=0 TO 1000:NEXT
270 GOTO 110
```

それではプログラムを見ながら、かんたんに解説していきましょう。

110　Ｊというハコを用意し、その中にＩという数字を入れます。コンピュータは、あとでこのＪのハコの中を見てカウントすることになります。

120～230　それぞれの転送元ページ指定に注目しましょう。Ｊになっていますね。今、Ｊの中はＩという数字が入っているわけですから、コンピュータはページＩから画像転送をします。

240　ＪのハコのＩ中にＩをたしています（Ｊの中はＩが入っていたわけですから、Ｉたして、Ｊの中は2になります）。

250　もし（ＩＦ）、Ｊのハコの中の数字が4より下（Ｊ＜4）ならば（THEN）、行120にもどりなさい（GOTO120）、ということです。Ｊの中身が4以上だったら無視して次の行にいきます。今、Ｊのハコの中は2ですから、行120にもどり、こんどはページ2から転送するわけです。そして行240で、2にＩたしてＪは3になり、行250の条件分岐で、また行120にもどる。ページ3からの転送作業がおわったらＪは4になり、行250で条件無視して行260へ。

という行程です。わかりますか？

イラストで見る条件分岐の流れ

110

彼は、Ｊという名のついたハコを用意し（ハコの名はＡでもＢでも本当はかまわないんだけどね）、その中に荷物をＩつ入れて次の行に送る仕事をしています。

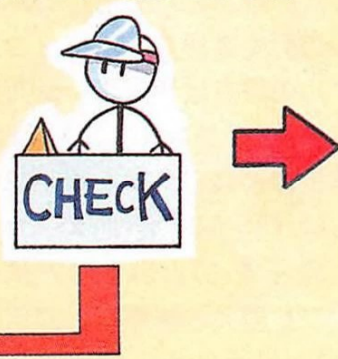
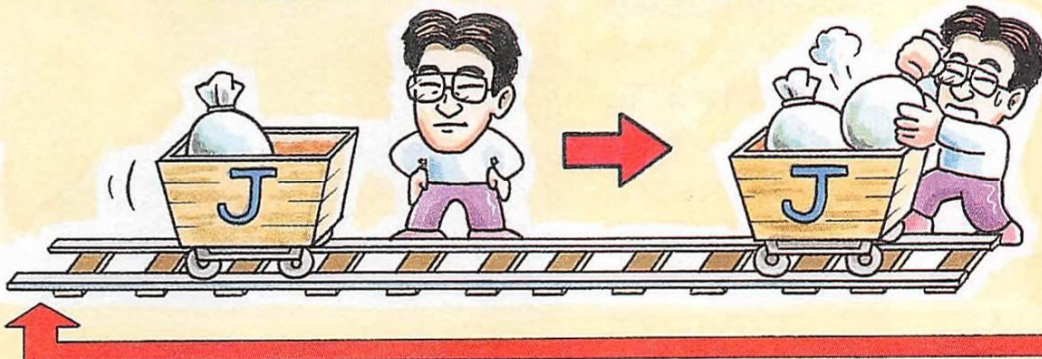
120～230

ここの彼は、ハコの中身の数をそのまま転送元にわりあてててアニメーション作業の命令をだしています。Ｉつ入っているので、ページＩから転送しろというわけです。

240

彼は、送られてきたハコの中に、もうＩつ荷物を追加する人です。ハコの中に荷物がいくつ入っていようと、かならず1個だけを追加する律儀な人です。

250

ここは、ハコの中に入っている荷物の数をチェックする税関です。荷物が4より少なければ、行120に強制送還させられます。4以上ならすんなりと通してくれます。

メーションでしたが、動きがよくできていました。次に静岡県のしじゃもファイターズクンの作品で『とーがらし』。なんだかよくわらない感覚のアニメーションですが、それが笑えました（北根編集長がイチオシしていた）。この2作品は、このままボツにするのもおしかったので、次回に保留ということにします。ごめんなさい（本当は今回全部採用したかったよ～）。

1学期のしめくくりとして、夏休みの宿題を出すよ～

1学期の授業はこれでおしまい。ふりかえってみれば、この1学期はじつに基本的なことをお教えした。キミたちもかんたんなアニメーションなら作れるようになったんじゃないかな？　それではいきなり夏休みの宿題を出しちゃうよ～。宿題はいままで習ったことを思いだしてアニメーションを作ること。べつに目パチロパクだけでもOKだけど、もっと手のこんだアニメーションならもっとOKだ。キミたちがどのくらい理解してくれたのかボクも知りたい。あて先は投稿と同じ、Mファンの紙芝居＆動画教室・宿題の係まで。学習結果をお待ちもうす！　それじゃ、また。　（中津）

⬆目パチロパクだけじゃなく、できれば楽しいアニメーションを作ってほしいな

☆「紙芝居＆動画教室」への投稿募集中！　応募要項は81ページ

紙芝居&動画教室

今回のPOINT
キャラクタと背景を合成させてみよう

2学期最初の授業は合成の基本から始めたいと思います。2つの絵を重ねるにはどうすればよいのでしょうか？　いよいよ本格的にむずかしくなっていきますので、じっくり読んで研究してくださいね。

➔P108の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

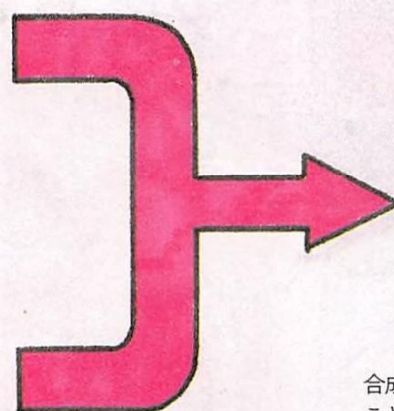
合成のあらまし

やっ／　中津です。夏休みは十分に遊びましたか？　私は仕事でした。シクシク……。

それはさておいて、新学期。今回のテーマは「CG合成」の第1回。パチパチパチ／　今までやってきたものから、さらにステップアップして高度なテクニックに入っていきます。気を引きしめてかかりましょう。このテクニックをものにすれば、基本レベルはクリアも同然。紙芝居倶楽部に掲載される日もグッと近くなるぞ〜。

それではさっそく授業に入りましょう。まずは合成のあらましから。そもそもCG合成とは何かというと、1枚のCGと、もう1枚のCGを重ね合わせることです。重ねることで得られる利点は、うまくやればCGをぐっと節約することができます。

具体的にどういうことかというと、たとえば木が描かれた背景CGがあって、その前に動く人物を合成すれば木の前で人物が動いているCGができますよね。これは、背景がついたキャラの動きの絵をたくさん描くよりも、背景1枚とキャラの動きだけを描いたほうが経済的です。でも実際やってみると、口でいっているほど単純なものではないんですけどね。

今回つけたサンプルプログラムは、あんまり複雑にしてもわかりにくいと思ったので、背景1枚と人物6枚でやっています。人物はいつものフレイでやりますので、付録ディスクのほうで見てくださいね。

重ね合わせの原理

合成とは、2枚のCGを重ねて1枚の絵にすることをいいます。ふつう、2枚の絵を重ねると、下にくる絵がかくれてしまいますが、上に重ねる絵（この場合は人物）のまわりを透明にしておけば、アニメセルのような重ねもできるのです。

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2版の『Xak』でデビュー。以後『FRAY』、『幻影都市』など、次々と手掛けたプログラマ&ゲームデザイナー。また、ときどきほかのMSX作品を手伝っているので、MSXにはなじみ深い人でもある。

AFTER SCHOOL 紙芝居倶楽部 〜投稿作品より〜 その1

【郵便で〜す／】

青森県／Fly☆Duck（28歳）
SCREEN5・全画面サイズ

今回の投稿作品は、全3本採用しました。まずそのうちの1つ『郵便で〜す／』です。この作品は、運動会のパン食い競走をパロったもので、ひじょうによくできています。今の時期も運動会の季節なのでピッタンコですね。見せ方や、プログラム的に見てもよくできていて、なにもいうことがありません。でもただ1つ、かけ足の動画がイマイチだったところが残念です。動きの研究をしてくださいね。

作者より一言

このアニメーションは、2年前にいちど挫折したものを引っぱり出してきて完成させたものです。オンメモリで動作するはずでしたが、色々いじっているうちにメモリに収まりきらなくなってしまいました。プログラムはパレット関係をのぞいてほとんどBASICで、MSX2／2＋でもそれほど速度落ちしないで動きます。解析してみるのもいいかもしれません。でもかなり変なことをやっているので、初心者にはむずかしいかも。　青森県・Fly☆Duck

合成時の問題と解法

さて、合成のプログラムを説明する前に、合成の仕組みと問題点を説明しましょう。

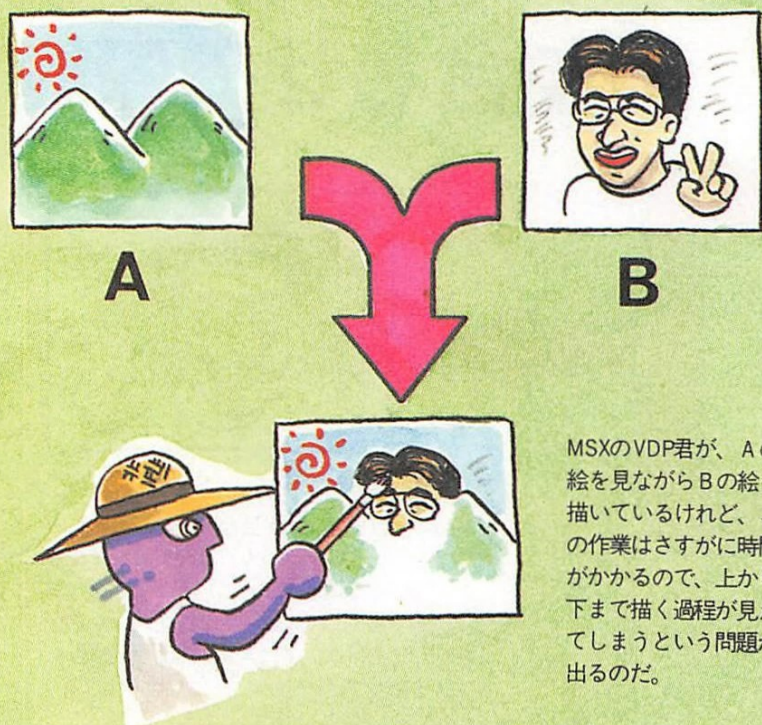
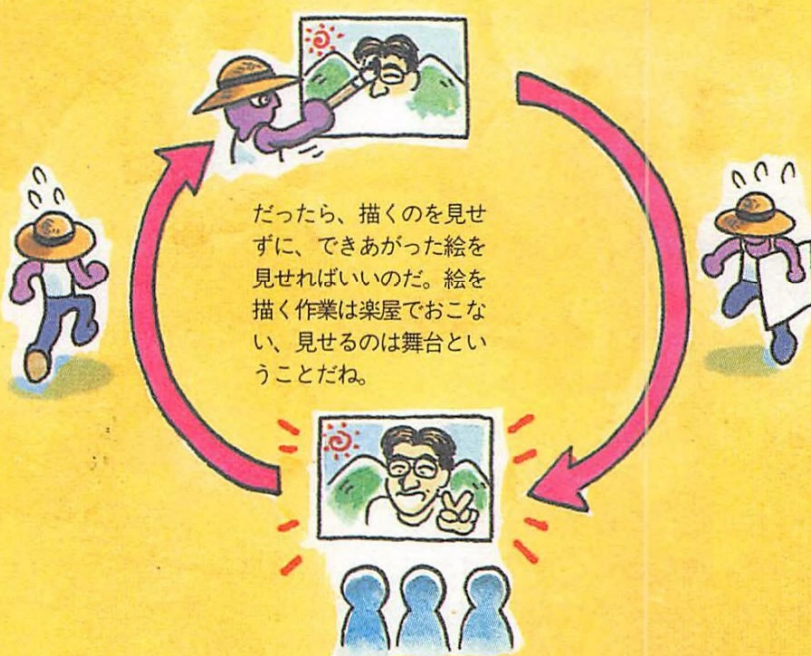
CGの合成は、もうおなじみ「COPY命令」を使って行なわれます。このCOPY命令は、MSXの画像チップであるVDPを動作させ、CGをちがうページに転送させる命令であることは覚えていますよね。

そのCOPY命令の中に、ロジカルオペレーションと呼ばれるものがあり、その中の一つに「TPSET」という命令が含まれています。

この命令は、CGの上に別のCGを、カラーコード0(透明色)の部分を除いて転送することができます。すなわち、16色あるMSXの色のうち、カラーコード0でぬられている色だけは転送されません。カラーコード0は、透明なセル板と同じと考えてくださいね。

さて、これを使えば2枚のCGを合成する事ができるのですが、ここで問題が1つ。この命令でのVDPの処理はいかんせん遅いのです。そのため、重ね合わせ作業を舞台(表示画面)でおこなうと、2つのCGを重ね合わせている工程が、目に見えてしまうという問題がおこります。どういうふうに見えるというと、チラチラとチラついているふうになります。

その問題を解決するための方法がバッファを使うこと。バッファとは、一種の作業場所という意味で、いったん見えない場所でデータを加工してたくわえる場所のことをいうのです。

この場合なら、楽屋をバッファとして、その中で合成作業をおこなってしまい、合成後のCGを舞台に転送表示させるという方法で、チラつきを感じさせない合成CG表示を行うことができるというわけです。かんたんにいえば、楽屋の中にある絵を楽屋の中で合成し、最後に舞台に転送するということです。

AFTER SCHOOL

紙芝居倶楽部 その2 ～投稿作品より～

【忍(シノビ)】

兵庫県/COMEON太郎(17歳)
SCREEN5・9分の1画面サイズ

小さな赤と青の忍者が、画面せましと戦いをくりひろげるこの作品。グラフィックにとりわけこっているわけではないのですが、こまかい動きがとてもすばらしいので採用しました。小さいながらもキャラクタの動きはじつになめらかで、見ていて気持ちがいいです。付録ディスクで見る人は、ぜひこの動きに注目しましょう。ターボRユーザーの人は、標準モードでお楽しみくださいね。

作者より一言

この作品は、動きがスムーズで気持ちイイと思う。けど、プログラムはきたない、最後のほうの動きがあらい、オチがありがちなどツメがあまいです。僕は絵がヘタなので1色キャラでごまかしてます。元絵は自作のへボいツールで描いたので、ごちゃごちゃしてます。このツールでのノウハウを生かしてSCCREEN12用多色刷キャラクタエディタ「CHED」を作りました。また送るので宜しくお願い申しあげたく存じ上げ奉り候。
兵庫県・COMEON太郎

サンプルプログラムで、とりあえず合成体験

今回右のサンプルプログラムで、はじめて使っているGOSUB命令を説明しておかないといけませんね。

GOSUB命令は指定された行番号にジャンプします。ジャンプ命令といえばGOTO命令とにていますが、GOSUB命令は、ジャンプ先でRETURNという表記があれば、GOSUBが出された次の命令にもどってくる性質があります。この飛び先を一般にサブルーチンと呼びます。右のサンプルプログラムでは、CGの座標をサブルーチンにわたして、合成、表示してもらうのに使っています。今回の解説は、プログラムの流れといっしょに説明しているので参考にしてくださいね。

それでは、じっさいに組みこむときのアドバイスをば。

まず、1パターンの合成したCGを表示させるためには、3回のCOPY命令を使うということです。わかりますか?

それでは説明しましょう。ページ0を表示画面、ページ1にキャラデータ、ページ2に背景データ、ページ3をバッファに使うとします。

すると、まずページ2の背景データをバッファであるページ3に転送します。これでCOPY命令1回。次にページ1にあるキャラのデータを、ページ3に転送して合成(COPY命令の表記にTPSETをつける)。これで2回。そして、バッファで合成された絵を表示画面に転送するのに1回で計3回使うのです。

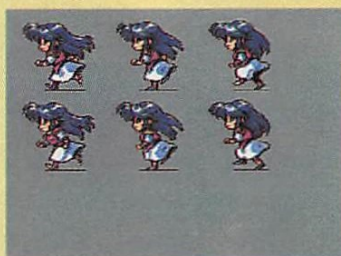
いかがですか?1コマの絵を表示させるのに3回もの転送命令を使うので、プログラムが長くなりがちです。GOSUBが必要なのがわかるでしょう。

さて、次回は合成テクニックの第2回。背景を動かすCG合成についての解説を予定してます。できれば、多重スクロールも解説したいなと考えています。

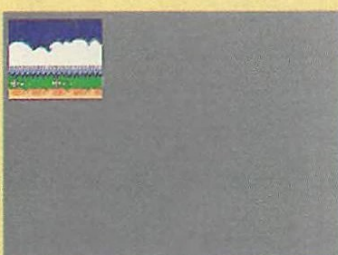
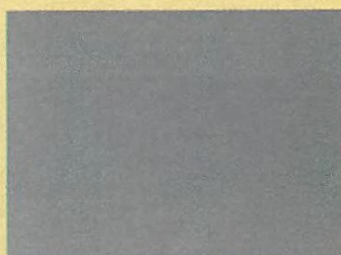
そんなこんなで、また次回/ 投稿作品をお待ちしていまーす。　(中津)

【VRAMの割り当て】

ページ0(表示画面)

ページ0は表示画面に使用されます。合成された絵が表示される、いわば舞台にあたります

ページ1

ページ1にはフレイの走るパターンが入っています。フレイのまわりのねずみ色の背景は透明色です。

ページ2

今回ページ2には背景だけのグラフィックデータが入っています。このデータをページ3に転送するのです

ページ3

ページ3は合成作業をおこなうバッファに使用します。合成したらページ0に転送をおこなうのです

SAMPLE3. BAS

```
10 SCREEN 5:COLOR 15,0,0:CLS
20 SET PAGE 0,1
30 BLOAD "f5.ge5",S
40 SET PAGE 0,2
50 BLOAD"f5-2.ge5",S
60 COLOR=RESTORE
70 Y=0
80 X=0:GOSUB 160
90 X=72:GOSUB 160
100 X=144:GOSUB 160
110 Y=72
120 X=0:GOSUB 160
130 X=72:GOSUB 160
140 X=144:GOSUB 160
150 GOTO 70
160 COPY (0,0)-(71,71),2 TO (0,0),3
170 COPY (X,Y)-(X+71,Y+71),1 TO (0,0),3,
TPSET
180 COPY (0,0)-(71,71),3 TO (97,72),0
190 FOR I=0 TO 100:NEXT
200 RETURN
```

それではプログラムを順をおって説明します。

10〜60　初期設定。

70〜80　Xに0、Yに0を入れて行160へ。

160　ページ2から背景グラフィックをページ3上に転送しています。

170　ページ1よりフレイのアニメパターンの1コマ目をページ3に転送。ページ3にはすでに背景データが入っているので、ここでこのフレイの絵と合成されることになります。TPSETがあるのに注目しましょう。

180　ページ3で合成された絵を舞台に転送。これで1コマ目の作業はおわり。

200　RETURNがあるので、さっきGOSUB命令が使われた次の命令(行90)にジャンプ。この作業のくりかえしで、フレイの合成アニメーションをやっているのです。

AFTER SCHOOL

紙芝居倶楽部 その3

~投稿作品より~

【とーがらし】

静岡県/ししゃもファイターズ
SCREEN5・全画面サイズ

とうがらしがあって、それをどうしようかと悩んだすえ、ピンと思いつく。はなぢ……。このわけのわからない感覚がおもしろくて採用です。FM音源で音楽がなりますので、MSX−MUSICのあるMSXで見てくださいね。またこの作品は、テクノポリス増刊・同人ソフト大全集に掲載された画面のフェードイン・アウトのプログラムを使用して作られています。

作者より一言

文化祭で多少けずったモノを上映したら、わりとウケていたので「載るな」と思ってました。次回作も松井君が原作の「ゴルザベッチャ素敵フレンズ」です。とりあえず「とーがらし」をどうぞ。(ホルスタイン渡辺)/コンニチハ。ワタシニホンゴジョーズナイネ。ワタシノクニハロシアアルヨ。ワタシノオシショサンチャイコフスキーデガスヨロシク(サルスキー)/こんなくだらない話を次回も送らせていただきます。今後ともヨロシク(クマンパイア松井)

☆「紙芝居&動画教室」への投稿募集中! 応募要項は81ページ

紙芝居&動画教室

今回のPOINT
背景をスクロールさせるテクニック

背景スクロールに挑戦！　むずかしそうなひびきがするけど、じつはとってもかんたんなもの。いままで習ったCOPY～でできるんだからね。今回は、そのスクロールをばっちり理解してもらっちゃうぞ。

→P108の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

かんたんなスクロールのしかた

お待たせしました！　毎度おなじみの中津です。年末ソフトのマスターアップもさし迫り、グルグル目が回っている今日このごろ、みな様はすこやかにおすごしでしょうか？　私は疲れてます。でもガンバルっス！

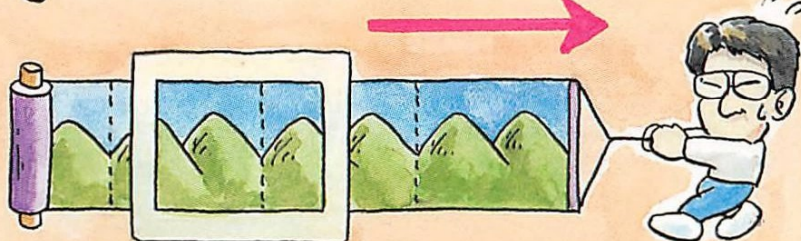
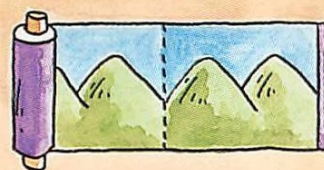
第2回目の「CG合成」は、背景スクロールに挑戦してみましょう。むずかしそうに見えますが、わかってみれば案外かんたんなものですよ。

それではまず、背景スクロールはどうやってするのかを考えてみましょう。

いつものフレイのサンプルを思い出してください。フレイが左に向かって走っていくと、背景はフレイに合わせて動き、左側が見えてこなければいけません。すなわち、左にだんだんずれていくことになります。

このずれがポイント！　じつはいつものCOPY命令を使い、ちょっとずつずらして転送することにより、流れる背景を作り出すことができるのです。

そのとき、1つ注意が必要です。ループしてる背景を流すときは、最低でも同じ背景画面を2画面分つなげておきます。1回目ずりずり～とそのまま流したあと、また同じ画面がやってきますね。そのとき、最初の画面にもどって転送をくりかえせば、永遠に背景は流れます。

スクロールを考える

たった1枚の背景画を横スクロールさせたいと思ったとき、どうすればよいのでしょうか。まず背景を横につなげてみます。そうすると、同じ背景の使いまわしで横に長い絵ができあがります。その絵を画面左から右へ流してみるのです。画面内では連続してその背景が流れているのでスクロールに見えるわけですね。

走るフレイとスクロール背景のひみつ

前回はフレイを背景の前で走らせるだけでした。背景は動かないのでフレイは足ぶみしているだけだったよね。今回の授業は背景も動かすということで、どうやって背景を動かすかをかるく考えてみましょう。それはかんたん。背景をちょっとずつずらして転送、そこにフレイの走る部分を合成すればOK。下の図はそれを表したものだよ。

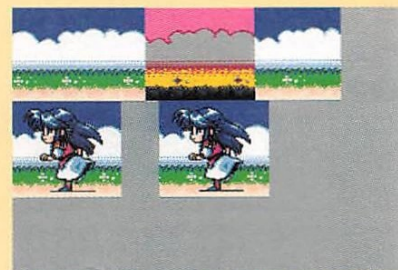
合　成

1コマ目

2コマ目

3コマ目

SAMPLE1.BASで一発理解！

それでもわからない人のために、付録ディスクに収録したサンプルプログラムから「SAMPLE1.BAS」を実行してみよう。これは、今回のメインサンプルプログラム「スクロール背景との合成」をわかりやすくしたもです。画面上が背景データ、その下の左がバッファ、右が表示画面を示してます。背景データが反転して、その部分を切り出しているというのがわかるでしょう。これによりスクロールの表現を作り出しています。

↑このサンプルを実行すれば一目瞭然。転送部分が一目でわかるぞ～

プログラムの手順を考えよう

では、今回のサンプルを参考に、プログラムを組む手順というものを考えてみましょう。

今回も１コマの絵を表示するのに、COPY命令を全部で3回使います。

まず、背景データをバッファに転送します。これで１回使用しますね。

次に走るフレイの１コマ目をバッファにある背景に合成転送します。合成のしかたは前回教えました「TPSET」を使います。前回この命令の使い方が説明不十分でしたので、下に復習コーナーを設けました。合成転送時のTPSETの書式を確認してください。

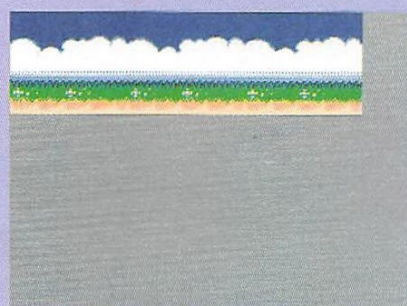
お話をもどしましょう。フレイのアニメデータをバッファに転送することで、COPY命令を2回使ったことになりますね。

これで合成された１コマ目の絵が完成。これをバッファから、表示画面に転送しましょう。これで、合計3回COPY命令を使うのです。わかりますか？

今回は、背景をスクロールさせることですので2コマ目からは、背景だけ左にずらして転送させます。ちなみにサンプルでは、フレイの歩幅に合わせて8ドットずつずらしていますよ。

↑今回のサンプルは、前回の走るフレイに背景をスクロールさせてみます

VRAMの割り当て

ページØ（表示画面）

ページ1

ページ2

ページ3（バッファ）

☆考え方☆

上の本文の解説をもっとわかりやすく、写真つきで説明してみましょう。まず、このページ右上の「VRAMの割り当て」を見てください。今回のサンプルプログラムのVRAMの割り当ては、以上のようにします。表示にページ０、フレイのアニメパーツをしまっているページ１、背景データをしまっているページ２、そしてバッファに使用するページ３です。それでは、これを頭に入れた上で、解説に入っていきましょう。

①転送

ここはページ３（バッファ）内です。まず、ページ２から背景データが送られてきました。COPY命令で、ページ２内にある背景データをバッファであるページ３に転送しているわけですね。

②合成転送

次に、ページ１よりフレイのアニメパーツの１部が転送されて、背景データの上に合成されました。合成は、COPY命令にTPSETという文が加えられてます。下の復習コーナーも参考にしてください。

③表示

ここは表示画面（ページ０）です。②でできた合成された絵が転送されてきました。これで１コマ目の作業はおしまいです。①にもどって、こんどは背景データをちょっとずらして、作業をくりかえします。

●復習コーナー● 合成は、TPSETと書き足すだけ

前号が出たあとのある日、編集部に紙芝居あての質問電話がきた。「合成表示をしたいんですけど、どうやったらよいのですか？」。あれれ？　わからなかったかな？　前回は本文中でしかその説明をしなかったから、具体的なプログラムの書き方がわからなかった人がほかにいるかもしれない。そこで復習コーナー！　合成転送の使い方をもういちど学んでみよう。COPY命令の中に含まれるTPSET、プログラムに組みこむときは、右のように最後にTPSETと書きくわえるだけでOKなのさ！

例文

```
COPY (Ø,Ø)-(71,71),1 TO (97,7Ø),3,TPSET
```

訳：ページ１の中にある、座標Ø，Øから71，71までの四角く囲った部分を、ページ３上の座標97，7Øにカラーコード Ø（透明色）の部分をのぞいて転送しなさい

サンプルプログラムで学習しよう

それでは、サンプルプログラムを解説しましょう。みなさんがあまりなれていないGOSUB命令についても説明します。

10〜60　初期化とCG読みこみ。ただし背景CGの元データは1枚絵ですので、ここでCOPY命令を使い、3枚分同じ背景をページ2内にならべています。

70　ハコを用意。ハコBの中は合成バッファのページ。すなわちページ3。ハコIXの中は、背景CGの転送元X座標。今は144という数が入ってます。

100〜190　メインルーチン。全部で6コマもあるフレイの走るアニメ制御を行なっています。ハコのXとYは、フレイの転送元の座標を表しています。そしてGOSUB命令を使い、1コマ1コマ行500にJUMPしています。このとき変化していることは、フレイの座標だけ。すなわち、ハコX、Yの中のにある数字だけです。

500　ページ2から背景CGをページ3のバッファに転送。ただし、転送元のX座標がハコIXの数をベースに計算されていることに注意してください。ハコIXの中の数が変われば(行540で変えている)、背景の位置がずれて転送されるのです。

510　フレイをページ1からページ3のバッファに合成転送(ハコXとYはその座標)。TPSETが書かれているのに注目。

520　ページ3のバッファからページ0の舞台に転送して、1コマ分の処理は終了。

540〜550　ハコIXの中の数字を変化させてます。ただし、数が0以下になったら、初期値のX座標に戻すという条件分岐があるのに注意しましょう。

590　RETURNで、JUMPしてきたGOSUBの次の命令に戻ります。GOSUBとGOTOのちがいはこれです。どこからでも呼び出せて必ずGOSUBの次の命令に戻る。これが処理の共通化を可能にします。メリットはプログラムが短かくなり単純化されるので、プログラム全体をチェックしやすくなる。共通化するとかんたんにデバックポイントをしぼれるなど。ぜひ身につけてください。

と、プログラムについては以上です。

次回あたりから(次回といい切れないとこがポイント)、高度な質問に対して答える常連さんコーナーなんてものを作ってみたいと考えてます。

紙芝居&動画に関する、あんなことやこんなことを説明したらいいんじゃないか、などのお手紙をお待ちしております。プロが使っているテクニックでも、出来るだけ解説するつもりです。特にいつも投稿してくるハイレベルなみなさんのするどいつっこみを期待しています。よろしくお願いしますね。

もちろん、投稿も随時大募集中。こっちもガンガン応募してよね。待ってるよ！　(中津)

SAMPLE2.BAS

```
10 SCREEN 5:COLOR 15,0,0:CLS
20 SET PAGE 0,1
30 BLOAD "f5.ge5",S
40 COLOR=RESTORE
50 COPY "f5.dat" TO (0,0),2
51 COPY (0,0)-(71,71),2 TO (72,0),2
52 COPY (72,0)-(143,71),2 TO (144,0),2
60 SET PAGE 0,0
70 B=3:IX=144
100 Y=0
110 X=0:GOSUB 500
120 X=72:GOSUB 500
130 X=144:GOSUB 500
140 Y=72
150 X=0:GOSUB 500
160 X=72:GOSUB 500
170 X=144:GOSUB 500
190 GOTO 100
500 COPY (IX,0)-(IX+71,71),2 TO (0,0),B
510 COPY (X,Y)-(X+71,Y+71),1 TO (0,0),B,
TPSET
520 COPY (0,0)-(71,71),B TO (97,72),0
530 FOR I=0 TO 20:NEXT
540 IX=IX-8
550 IF IX<1 THEN IX=144
590 RETURN
```

AFTER SCHOOL

紙芝居倶楽部
〜投稿作品より〜

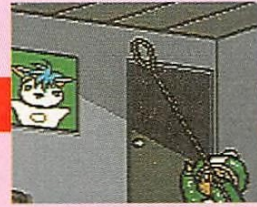
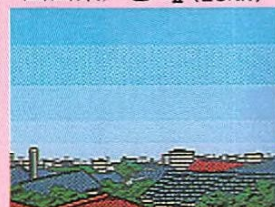
[a sad memory]

山口県／S-II(20歳)　SCREEN5・4分の1画面サイズ

今回の採用者はS−IIクンの1作品だけ。でも、とてもすばらしい作品です。小人が寝ている女性の顔にラクガキをするというストーリー。オーバーなアクションと、ほのぼのとしたお話がマル。さらにこの作品、なによりも、よくメモリに入ったと思わせるデータ量。背景をプログラムで作ったテクニックが決め手ですね。

作者より一言

MSXでアニメをする場合、VRAMをいかに上手に使うかがコツです。だから私自身、アニメの途中でディスクアクセスすることはタブーとしております。そういう理由で初心者から中級者へステップアップしたい人は、プログラムではなく、裏ページを参考にしたほうがよいかもしれないよん。

山口県・S−II

紙芝居&動画教室

今回のPOINT

多重スクロールに挑戦しよう。

今回は、多重スクロールの基本をやってみます。でも少々むずかしいかな? COPY〜の応用もここまで来ると少々複雑。けど、やってみると何とかなるものです。サンプルをいろいろといじってみてね。

➜P108の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

多重スクロールを考える

「CG合成」もいよいよファイナル! どうも、中津です。PCエンジン版のフレイのデバッグもほぼ終り、どうやら無事に年もこせたようです(原稿を書いている時点では予定。PCエンジンFAN等の雑誌に載るのは1月末あたりからかな? CMでした〜)。

さて、第3回目になる「CG合成」は、手前のCGをスクロールさせ、多重スクロールを見せるという基本テクニックです。計算が少々複雑なので面食らうかも知れませんが、やっていることはむずかしいことではありませんから安心してください。それでは授業に入りましょう。

今回は、いつものフレイのサンプルで、フレイの前を木が横切るものを作りましょう。木は左からゆっくり動いてきて、フレイの手前を流れて右側に消えていきます。

このとき、表示されている画面の中で木は、左側部分が見えないとき、全体が見えているとき、右側部分が見えないときの3パターンがあるのがわかりますね。

今回も、もちろんCOPY命令を使ってプログラムを組みますが、この見えてない部分がある木をうまく転送するにはどうすればよいのでしょうか?

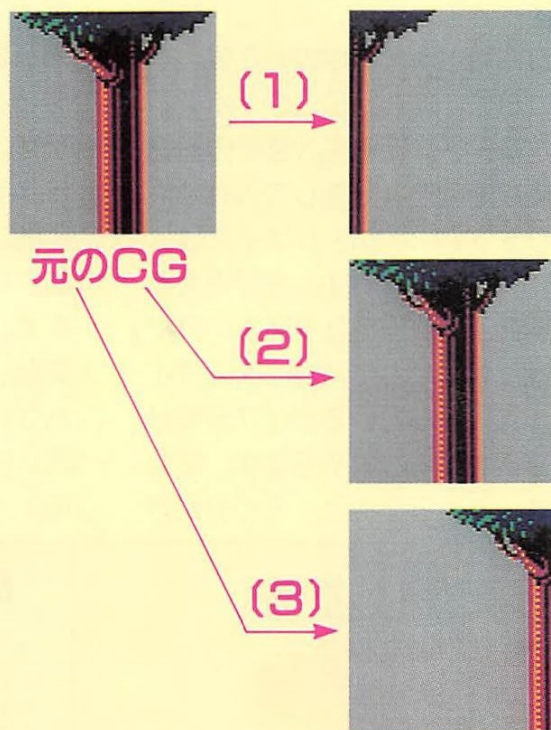
とうぜんCOPY命令はそんなことをつごうよく処理してくれるわけではありません。ここはプログラムできちっとおぎなってあげましょう。考え方としては、木を表示する位置に合わせて、担当するプログラムを分けてしまうのがかんたんなやり方です。

木の左側が見えないとき、転送する元のCGのX座標を右にずらすことで見えない部分を転送するのをふせぎ、あたかも左が消えているかのように見せるCOPYを実行します。逆に、右側が見えないとき、転送するCGのサイズを小さくすることで、あたかも右が消えているかのように見せるCOPYを実行すれば良いのです。それ以外は、全て見えているときなのでふつうにCOPYします。

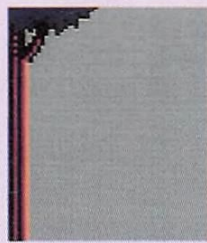
これらのずれやサイズの計算は、表示画面のサイズと木のCGのサイズを元に簡単に計算することが可能です。くわしくは、次ページのサンプルプログラムといっしょに説明していきましょう。

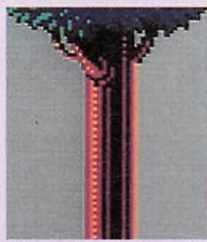
今回の目的は?

●今回はフレイの手前を木が横切ります。これだけでも複雑なので、今回背景はスクロールさせません

1本の木を左から右へ流してみる

1本の木を左から右に流すには「木の左側が見えないとき」、「木の全体が見えているとき」、「木の右側が見えないとき」の3パターンがあります。これをプログラムするとき、どのようにすればいいのでしょう。それは、木の表示座標をIF文でチェックし、仕事をふり分けてやるのです(くわしくは次ページ)。木の左側が見えないときは表示座標(木の左側の座標)が表示画面より左なとき、右側が見えないのは、木の右側の座標が表示画面をはみ出しているときです。1本の木をCOPY命令でうまく切り取り、あたかもスクロールさせているように見せるのがコツです。

解法

COPYを場合によって使い分けるのだ

左側を転送するCOPY

1本の木の右側部分を切りとって転送する。

全体を転送するCOPY

木全体を転送。転送先座標をずらしてスクロールを表現する。

右側を転送するCOPY

1本の木の左側部分を切りとって転送する。

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2版の『Xak』でデビュー。以後『FRAY』、『幻影都市』など、次々と手掛けたプログラマ&ゲームデザイナー。また、ときどきほかのMSX作品を手伝っているので、MSXにはなじみ深い人でもある。

プログラムを組んでみよう

それでは、このページの右下にあるサンプルプログラムを解説していきましょう。計算が複雑になってきたためハコがいっぱい出てきますが、リストとてらし合わせながら、ゆっくり読んでいきましょう。あせらずにいきましょうね。

10〜60　初期化とCG読みこみ。ファイル"F5．GE5"がフレイの走りCG。ページ１に読みこみます。ファイル"F5．DAT"が背景と木のCGで、部分セーブされています。このデータはページ2に入ります。

70　ハコBは合成のバッファのページ(ページ3)。ハコPXは木のCGの初期の転送先X座標。ハコＩＸは変化する転送先X座標で、ＰＸを入れて初期化。

100〜190　メインルーチン。走るフレイのアニメの制御を行なっています。ハコXとYは、フレイの転送元の座標を表しています。GOSUB命令で、行500に行き、１コマの処理をします。

510　ページ2から背景CGをページ3のバッファに転送。

520　フレイをページ１からページ3のバッファに合成転送(ハコXとYはその座標)。ＴＰＳＥＴが合成のための呪文であることはもうわかりますね。

540　ハコＩＸが０以下なら、左側が消えたCGを転送するための計算を行ないます(ＩＦ文を使って確かめています)。ハコＢＳに転送元X座標の補正値。ハコSZに転送元CGのサイズ。ハコPTに転送先X座標の補正値をそれぞれ計算して入れてやります(木のCGのサイズは、70×70ドットです)。左が消える場合、ハコBSは見えない分だけ、すなわち、ハコＩＸのマイナスになっているサイズ分だけ座標をずらしてやります。そして、ハコSZには、その分だけマイナスしたCGサイズを入れます。ハコPTは、左端ですので０を入れます。以上計算が終れば、行600にジャンプします。

550　ハコＩＸを元に右側の座標を計算し、右側が消えているかチェックします(ここもＩＦ文でチェック)。70は木のCGサイズ。72は、画面のCGサイズです。右が消える場合、ハコBSは０で補正の必要なし、ハコSZには、右側にはみ出した分だけマイナスしたCGサイズを入れます(サンプルでは式を展開してあります)。ハコPTは、現在のハコＩＸと同じ座標になります。以上計算が終れば、行600にジャンプします。

560　基本的にそのまま処理します。ハコBSは０で補正の必要なし、ハコSZも70の固定、ハコPTは、現在のハコＩＸと同じ座標になります。

600　サイズが０以下のものをキャンセルしています(表示しないため)。

610　ページ2にある木をページ3のバッファに合成転送しています。このとき、ハコBS、SZ、PTを元に転送元や転送先の座標を計算して転送する大きさなどを変化させていることに注目してください。ここで、右側、左側の消える部分を最終的に処理しているのです。

650　ページ3のバッファからページ０の舞台に転送して、１コマ分の処理は終了。

670〜680　スクロールの規準座標のハコＩＸを変化させる。ただし、72以上になったら、初期値のPX座標に戻します。

690　ＲＥＴＵＲＮで、ジャンプしてきたGOSUBの次の命令に戻ります。

これがサンプルプログラムの流れです。複雑になってきましたが、わかりますか？

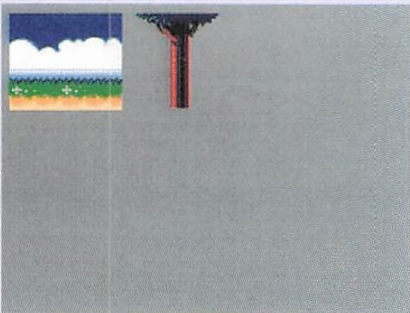
SAMPLE1で学習しよう

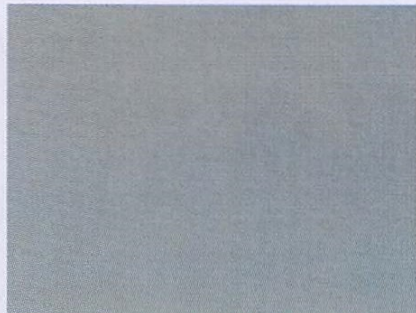
VRAMの割り当て

今回のVRAMの割り当ては、写真のように割り当てます。ページ１にはいつものフレイの走るグラフィックデータ。先日質問電話があって「このグラフィックは１画面に入っているのですか？」とのこと。そうです、１画面で6コマのフレイのアニメパターンをしまっています。それをCOPY命令で切りとって転送しているわけです。ページ2には背景データと木のデータが、ページ3は合成用のバッファに割り当てます。

ページ1(F5.GE5)

↑いつものフレイのグラフィックデータはページ１内におきます。お約束ですね

ページ2(F5.DAT)

↑付録ディスクに入っているデータは、背景と木だけの部分セーブです。注意してね

ページ3

↑背景とフレイと木を合成するバッファ。合成完了したらページ０に転送します

SAMPLE1. BAS

```
10 SCREEN 5:COLOR 15,0,0:CLS
20 SET PAGE 0,1
30 BLOAD "f5.ge5",S
40 COLOR=RESTORE
50 COPY "f5.dat" TO (0,0),2
60 SET PAGE 0,0
70 B=3:PX=-70:IX=PX
100 Y=0:'ﾒｲﾝﾙｰﾁﾝ
110 X=0:GOSUB 500
120 X=72:GOSUB 500
130 X=144:GOSUB 500
140 Y=72
150 X=0:GOSUB 500
160 X=72:GOSUB 500
170 X=144:GOSUB 500
190 GOTO 100
500 'ｻﾌﾞﾙｰﾁﾝ
510 COPY (0,0)-(71,71),2 TO (0,0),B
520 COPY (X,Y)-(X+71,Y+71),1 TO (0,0),B,
TPSET
530 'こんかいのポイント きは,70*70ドットなので ちゅうい
540 IF IX<=0 THEN BS=-IX:SZ=70+IX:PT=0:G
OTO 600:'ひだりがわ
550 IF (IX+70)>=72 THEN BS=0:SZ=72-IX:PT
=IX:GOTO 600:'みぎがわ
560 BX=0:SZ=70-IX:PT=IX:GOTO 600:'ぜんたい
600 IF SZ<=0 THEN 650
610 COPY (72+BS,0)-(72+BS+SZ-1,69),2 TO
(PT,0),B,TPSET
650 COPY (0,0)-(71,71),B TO (97,72),0
660 FOR I=0 TO 20:NEXT:'ｳｴｲﾄ
670 IX=IX+5
680 IF IX>=72 THEN IX=PX
690 RETURN
```

ハイテクニックサポートコーナー　TIME関数の使い方

前回予告しましたハイテクニックコーナーです。ここでは、初級を卒業した投稿者のために、表ワザ、裏ワザのサポートをしていきたいと思います。今回は、投稿者のFly☆Duckクンの手紙をベースに、TIME関数の使い方を説明します。その手紙の内容はこうでした。

「プログラムを組むとき困るのが、ターボRとそれ以前の機種との実行速度の差です。機種を調べてウエイトを調節したり、強制的にZ80モードに切りかえてから実行したりといったプログラムもありますが、BASICのシステム変数であるTIME変数を使うという方法もあります。これまでも数秒単位のウエイトにTIME変数を使うといったプログラムはありましたが、コンマ何秒といった短い周期でも、ちょっと工夫すれば使えます。具体的には次のようにします。

１００　ＩＦ　ＴＩＭＥ＜Ｉ　ＴＨＥＮ　１００　ＥＬＳＥ　ＴＩＭＥ＝０（Ｉには１／60秒単位の数が入る）

行の最後でTIME＝０を実行し、ほかはなるべくＴＩＭＥをいじらないというのがミソです。これをカーソル移動ルーチンのFOR～NEXTによるウエイトのかわりに入れれば、機種に関係なくほどよい速度でカーソルを動かせます。どういう意味かというと、ＴＩＭＥは１／60秒単位で時間をカウントするタイマーで、時計のように自動的に時間をカウントしていくものです。このタイマーは０を代入することでクリアすることもできます。とすれば、TIMEをクリアしてから別の処理をした後、そのTIMEの時間を見れば、どれだけ時間が経過したかわかるので、ウエイトにも使えるではないかということです。」

かんたんにいうと、現在この教室で使っているウエイトがFOR～NEXTを使っているため、２＋とターボRで同様に動作しないので不便ではないか、かわりにＴＩＭＥ関数を使えば解決できるよ、というものです。

使う方法は、処理の後でＴＩＭＥが規定の時間に達するまでループさせて待つことでウエイトさせます。ループをぬければ再びＴＩＭＥを０にクリアしてやるのです。これはプロが使うウエイトと同じ方法で、じっさい、スーパーファミコンやPCエンジンなどのゲームでもタイミングの調整に使うテクニックなのです。もちろん、MSXのゲームでもこれを使っているゲームはたくさんあります。このテクニックはぜひ覚えて、色々なプログラムに使ってください。

それじゃ、また！

SAMPLE2を走らせてみよう！

このサンプルはTIME関数を使った実行例です。ターボRで確認してください。実行すると、まず０から10の数字が高速で表示されます。そのあとまた０から10の数字がこんどはゆっくり表示されます。最初の高速表示は、ＦＯＲ～ＮＥＸＴを使った実行例です（行100～行140）。その後のゆっくり表示はTIME関数を使っています（行200～行240）。このゆっくり表示は、Ｚ80モードで実行したときとスピードはかわりません。TIME関数を使ったウエイトだからですね。ということで今回はおしまい。次回もハイテクコーナーに、紙芝居＆動画教室に関する、こんな事教えて、こんな事説明したらいいんじゃないか、等のお手紙をお待ちしておりま～す。

SAMPLE2. BAS

```
100 CLS:'no1 tRだと スピードはやい
110 FOR C=0 TO 10
120 LOCATE 0,0:PRINT C
130 FOR I=0 TO 100:NEXT
140 NEXT
200 'no2 tRでも かわらないスピード
210 FOR C=0 TO 10
220 LOCATE 0,1:PRINT C
230 IF TIME<10 THEN 230 ELSE TIME=0
240 NEXT
```

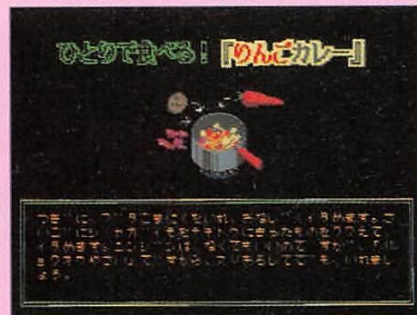
AFTER SCHOOL

紙芝居倶楽部 ～投稿作品より～

今回は投稿作品から２作品。日本の北の国、北海道の方２人からの投稿です。今回採用の２作品は、どちらもいままでのようなすごいアニメーションとかではなく、絵とお話だけの、いかにも「紙芝居」という作品です。たまにはこういった作品もいいですね。みなさんも負けずに投稿してくださいね。

【りんごカレー】

北海道／TOMすけ（43歳）　SCREEN５・全画面サイズ

りんごカレーの作り方を、紙芝居でお届けしているこの作品。ひとり暮らしの人は、これを見ながら自分で作ってみてはいかがでしょうか。おいしそうな野菜グラフィックと、けっこうにているサウンドで、とてもおもしろく紙芝居しています。聞けばファンダムの常連さんだそうで、またの投稿をぜひお待ちしております。

↑リンゴとカレーのナイスマッチ。はちみつは？　東は？

作者より一言

ファンダムではけっこう採用していただいていますが、こちらでは初採用です。この作品は、私が編集人をつとめるDMの目玉として、シリーズ化したものの第１弾です。MSXの新たな方向を模索した作品、といったら大げさかな？　ひとり暮らしのあなたにはうってつけのレシピソフト。今、第４作まで完成していますので、また味をしめて投稿させていただくかも知れません。ではまた。

（北海道・TOMすけ）

【私説三国志＝おみくじの巻＝】

北海道／りん（16歳）　SCREEN５・４分の１画面サイズ

こちらも聞けば１年前、ゲーム十字軍の三国志イラストの常連さんだったそうで、さすがに絵がかわいらしいですね。紙芝居的コマまんが作品になっています。登場人物は三国志の曹操、劉備、孫権の３人。そんけんのボケがいい味だしています。イラストコーナーは終わってしまいましたが、今度は紙芝居でがんばってください。

↑ゲーム十字軍で一世を風びした、そんけんのボケが今ここに！

作者より一言

本当に採用なんですか？　私投稿されてたことも知らなかったんです（ないしょで投稿した犯人は私です：TOMすけ）。でもうれしいですね。私は４コママンガを描くときは「いきあたりばったり」なんで、ストーリーをはじめから考えたりしません。これも最初はおみくじではなかったんですよ。ふふふ。これからもそんけん達をかわいがってください。では、本当にありがとうございました。（かーっ、教科書ど～りの文だわ）。

（北海道・りん）

今月のPOINT／復習もかねて完全多重スクロールに挑戦

最近3DOも関って、いそがしい毎日を送っている、中津です。今回はもうひとつの背景スクロールのやりかたを紹介し、いよいよ多重スクロールの基本をしめくくりたいと思います。もちろんハイテクコーナーもありますよ〜。

→P92の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

もうひとつの背景スクロールのさせかた

どうも、中津です。

さて、いきなりCM。3DOでキャビンのゲームが出ます。雑誌でチェックよろしく。それと、PCエンジンのフレイは3月25日に発売です。末永、新田の両名も参加しているのでぜひ買ってほしいぞ〜。てなことで、授業をはじめましょう。今回は復習もかねて完全多重スクロールに挑戦してみます。

今回、背景の転送方法がちょっとむずかしくなります。いままでは1画面の背景を横に3つつなげ、ちょっとずつずらしながら転送してスクロールをしていました。今回は1画面の背景だけでスクロールさせます。さてその方法とは？

原理はいたってかんたん。1枚の背景データを縦に切って2枚にわけます。この切れた2枚の右側をA、左側をBとします。背景を右へずらすので、Aをだんだん大きく、Bを小さくなるように切り、表示画面上でAB逆にして貼っていけば右へスクロールして見えます。

こうして背景はずっとスクロールをさせることができるのです。文だけではわかりにくいので、下のイラストも参考にしてくださいね。

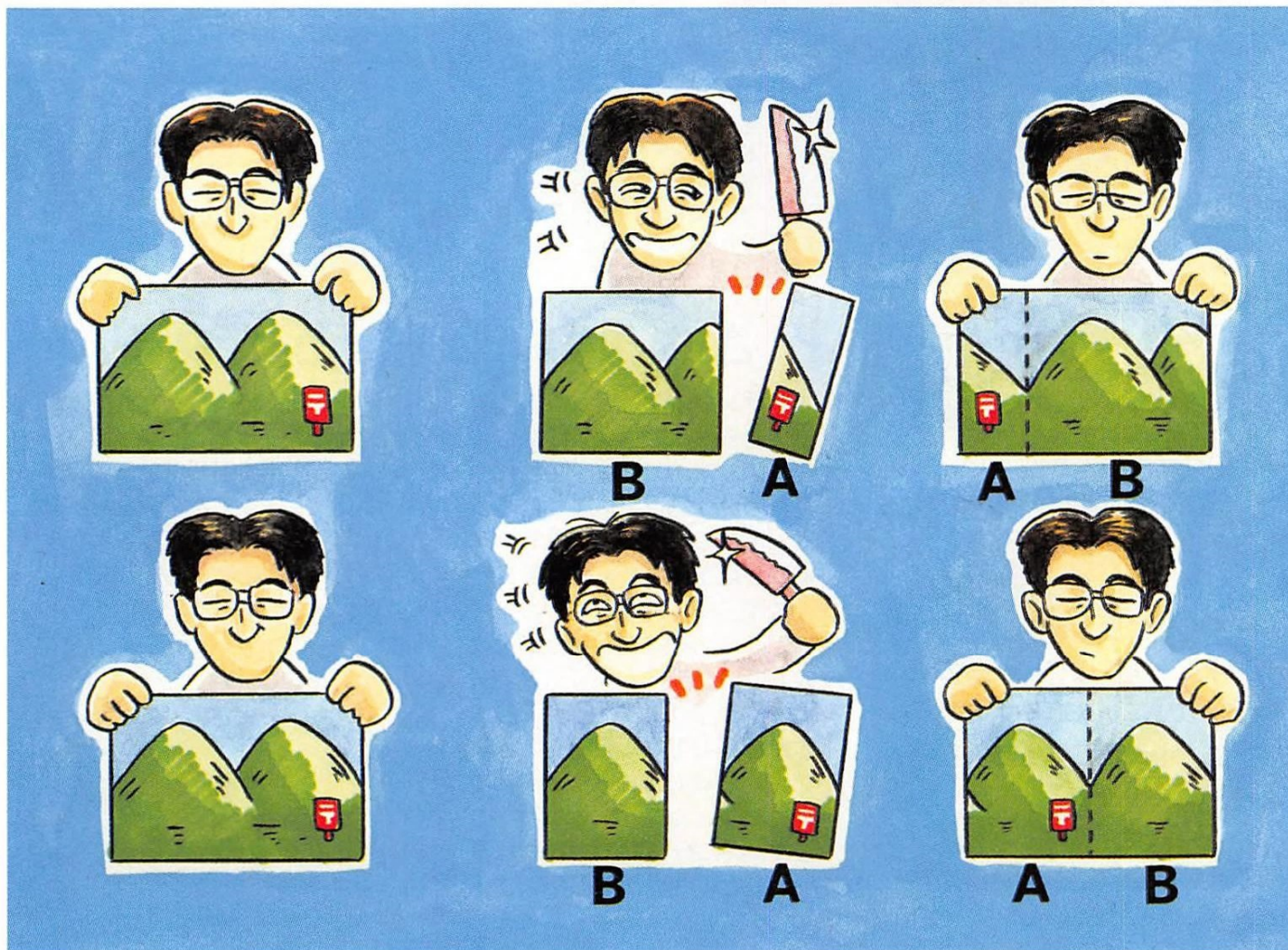
いままでは

いままでのやり方では、たった1画面しかない背景データをバッファ内で横にいくつもつなげ、それをちょっとずつずらしながらスクロールをしていました。

もう1つの方法

今回紹介するもう1つの方法とは、1画面の背景データを切りとって作るスクロール方法です。左のイラストの上段を見てみましょう。まず用意した1画面の背景データ。これを縦に切り2つにわけます。切り口はやや右よりです。小さいほうをA、大きいほうをBとし、表示の段階でAとBを逆につなげます。これで1コマ。同様に下段を見てみましょう。今度はちょうど半分にわけます。そしてまた表示の段階でAとBを逆につなげるのです。つなぎ終わった絵のそれぞれのポストの位置に注目しましょう。上と下の絵ではちゃんとずれているのがおわかりでしょう。こういうやり方でスクロールをさせる方法もあるのです。覚えましょう。

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の『ザーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ＆ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

サンプルプログラムで理解しよう

それでは、サンプルプログラムを解説しましょう。

10〜60　初期化とCG読みこみ

70　ハコBは合成のバッファのページ。ハコPXは木のCGの初期の転送先X座標。ハコIXは変化する転送先X座標で、PXを入れて初期化。ハコVXは背景のCGの初期の転送元X座標。

100〜190　メインルーチン。走るフレイのアニメの制御をおこなっています。ハコXとYはフレイの転送元の座標をあらわしています。GOSUB命令で行400に行き1コマの処理をします。

420　ハコVXが71以上であれば、行430の処理をパスします。

430　背景CGのハコVXであらわされる右側ブロックから表示画面の左側に転送します。

460　(ハコVX−1)が1より小さければ、行470の処理をパスします。

470　背景CGのハコVXであらわされる左側のブロックから、表示画面の右側(行430処理の余白)に転送します。

510　フレイをページ1からページ3のバッファに合成転送(ハコXとYはその座標)。TPSETが合成のための呪文。

540　ハコIXが0以下なら、左側が消えたCGを転送するための計算を行ないます。ハコBSに転送元X座標の補正値。ハコSZに転送元CGのサイズ。ハコPTに転送先X座標の補正値をそれぞれ計算して入れてやります(木のCGのサイズは、70×70ドットです)。左が消える場合、ハコBSは見えない分だけ、すなわちハコIXのマイナスになっているサイズ分だけ座標をずらしてやります。そして、ハコSZにはその分だけマイナスしたCGサイズを入れます。ハコPTは左端ですので0を入れます。

以上の計算終了後、行600にジャンプします。

550　ハコIXを元に右側の座標を計算し、右側が消えているかをチェックします。70は木のCGサイズ。72は画面のCGサイズです。右が消える場合、ハコBSは0で補正の必要なし、ハコSZには右側にはみ出した分だけマイナスしたCGサイズを入れます(サンプルでは式を展開してあります)。ハコPTは、現在のハコIXと同じ座標になります。

以上計算が終れば、行600にジャンプします。

560　基本的にそのまま処理します。ハコBSは0で補正の必要なし、ハコSZも70の固定、ハコPTは現在のハコIXと同じ座標になります。

600　サイズが0以下のものをキャンセルしています(表示しないため)。

610　ページ2にある木をページ3のバッファに合成転送しています。このときハコBS、SZ、PTを元に転送元や転送先の座標を計算して転送する大きさなどを変化させていることに注目してください。ここで右側、左側の消える部分を最終的に処理しているのです。

650　ページ3のバッファからページ0の舞台に転送して、1コマ分の処理は終了。

710　前回ハイテクコーナーで説明したウエイトを使用。よくわからなくてもこんな形で書けばよいと理解してください。

730〜740　スクロールの規準座標のハコIXを変化させる。ただし、72以上になったら初期値のPX座標に戻します。

760〜770　背景スクロールの規準座標のハコVXを変化させる。ただし、1未満になったら初期値の72に戻します。

790　RETURNで、ジャンプしてきたGOSUBの次の命令に戻ります。

むずかしいですか?

SAMPLE1.BAS

```
10 SCREEN 5:COLOR 15,0,0:CLS
20 SET PAGE 0,1
30 BLOAD "f5.ge5",S
40 COLOR=RESTORE
50 COPY "f5.dat" TO (0,0),2
60 SET PAGE 0,0
70 B=3:PX=-70:IX=PX:VX=72
100 Y=0:'== ﾒｲﾝﾙｰﾁﾝ ==
110 X=0:GOSUB 400
120 X=72:GOSUB 400
130 X=144:GOSUB 400
140 Y=72
150 X=0:GOSUB 400
160 X=72:GOSUB 400
170 X=144:GOSUB 400
190 GOTO 100
400 '== ｻﾌﾞﾙｰﾁﾝ == ちょっと こうどな はいけい てんそう
410 'Aぱーと てんそう
420 IF 71<=VX   THEN 450
430 COPY(VX,0)-(71,71),2 TO (0,0),B
450 'Bぱーと てんそう
460 IF (VX-1)<1 THEN 500
470 COPY(0,0)-(VX-1,71),2 TO (72-VX,0),B
500 '-- ﾌﾚｲ の ごうせい --
510 COPY (X,Y)-(X+71,Y+71),1 TO (0,0),B,
TPSET
530 '-- きのごうせい --
540 IF IX<=0 THEN BS=-IX:SZ=70+IX:PT=0:G
OTO 600:'ひだりがわ
550 IF (IX+70)>=72 THEN BS=0:SZ=72-IX:PT
=IX:GOTO 600:'みぎがわ
560 BX=0:SZ=70-IX:PT=IX:GOTO 600:'ぜんたい
600 IF SZ<=0 THEN 650
610 COPY (72+BS,0)-(72+BS+SZ-1,69),2 TO
(PT,0),B,TPSET
650 COPY (0,0)-(71,71),B TO (97,72),0
700 '-- ｳｴｲﾄ --
710 IF TIME<6 THEN 710 ELSE TIME=0
720 '-- き の ざひょう へんこう --
730 IX=IX+4
740 IF IX>=72 THEN IX=PX
750 '-- はいけい の ざひょう へんこう --
760 VX=VX-8
770 IF VX<1 THEN VX=72
790 RETURN
```

ここをいじると、もっとよくなる!

上のサンプルプログラムをそのまま走らせると、背景のスクロールより木のスクロールが遅く、見た目には遠近方が成り立ちません。そこでちゃんと見せるために、以下の行を自分で書きかえて走らせてみましょう。まず行730。これは木の移動座標を設定している部分です。ここの「4」という数字は、4ドットごとに木が移動しているということなので、12にしてみましょう。これで12ドットごとに木が移動します。次に行760です。ここは背景の移動座標を設定しています。ここの「8」を6と書きかえましょう。そしてRUN。どうです?遠近がきちんと成り立ち、フレイがちゃんと走っていますね。ほかにも木を2とか背景を1にしてRUNしてみてもおもしろいですよ。いろいろいじってみましょう。

⇧背景が流れ、木も流れ、そしてフレイが走る。多重合成わかったかな?

ハイテクニックサポートコーナーページ切りかえ

パチパチ／　第2回目のハイテクコーナーをお送りします。今回は、またまた投稿者のFly☆Duckクンからの手紙をベースに、ページ切りかえのテクニックを説明します。その手紙の内容はこうでした。

「アニメーションをおこなう場合、表示ページをページ0に固定して、他のページからグラフィックをコピーするのがふつうですが、これでは前景と背景の重ね合わせは裏画面でおこなう必要があり、COPY命令を3回実行しなければいけません。

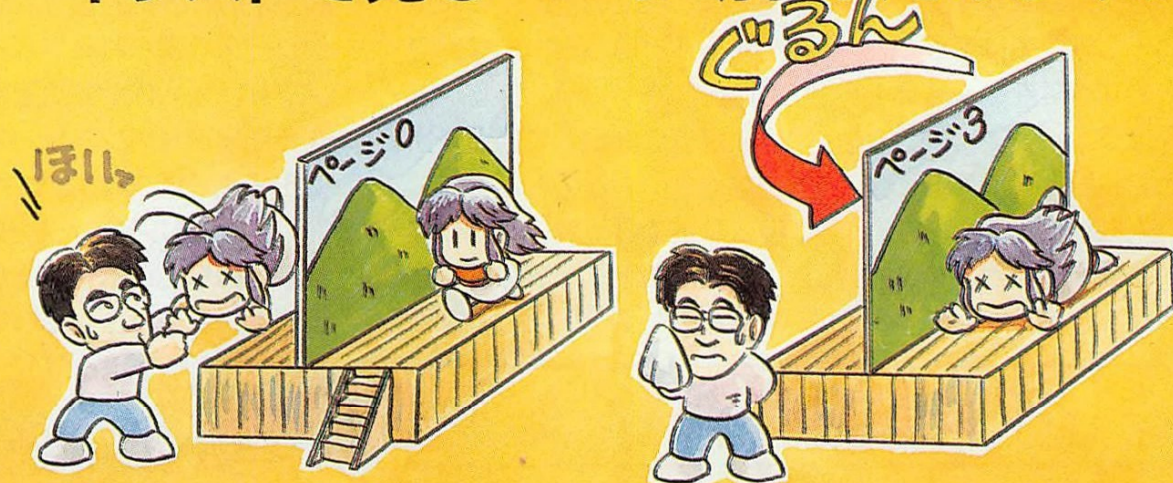
これに対し表示ページを2画面用意して、片方の画面を表示している間にもう片方に次の絵を用意し、交互に表示する方法にすればCOPYは2回ですみ、その分高速になります。Mファンで舞台と楽屋に例えて説明していますが、さしずめ回り舞台というところでしょうか。」

ということで、今回のSAMPLE1のプログラムをページ切りかえ(複数ページの交互表示をキャビンではこう呼んでいる)をしたらどうなるかというのがSAMPLE2のプログラムになります。

どうやっているかといいますと、表示画面をページ0と3としています。例えば、ページ0を表示している間にページ3で次のCGを合成し、終了したらページ3を表示画面に切りかえれば、COPYなしに表舞台にCGを表示することができます。

ページの切りかえはCOPYにくらべて何百倍も速いので文字どおり一瞬です。SAMPLE1と2をRUNしてもらえばわかると思います。

メリットとして、COPYを使って表に出さないので、どんなに合成されるCGが大きくても転送が見えることがない、COPYが1回へるので高速になる、ということ。デメリットとして2ページの画面を使うので、CGを置いておく場所が残り2ページになってしまう事が挙げられます。使うかどうかはその人の判断といえるでしょう。

最後に、キャビンのゲームでページ切りかえの例を説明しておきます。サークⅠはページ切り替えをしていません。なんせ、512ドットのモードですから、2ページも使ったらCGを置いておく場所が何も残らないんです。サークⅡやフレイは、ページ切りかえをしています。ちがいがわかりましたか？

いままではページ0を舞台、裏ページを楽屋としてきましたが、裏ページも舞台に利用できます。まずページ0を表示中にページ3で準備をします。次にページ3をそのまま舞台に回し、裏になったページ0で次の準備するやり方です。このほうが高速かつ効率がいいのです。

AFTER SCHOOL
紙芝居倶楽部
～今回の付録ディスクの収録内容について～

今回は投稿作品がこなかったので、付録ディスクには過去に採用された作品を収録しています。今回収録したのは、92年5月号で採用された「圭一愛の物語Ⅰ」と「圭一愛の物語Ⅱ」、そして92年7月号で採用された「圭一愛の物語Ⅲ」の3本パックです。作者はすべて三重県のFRIEVEです。投稿作品がなかったというのは悲しいことなので、みんなもっと投稿にはげんでほしいぞ。

そして授業のサンプルプログラム。付録ディスクに収録しているサンプルプログラムは計3本です。まず左ページの「SAMPLE1」。そして上で説明した「SAMPLE2」。最後の「SAMPLE3」は注意してください。これは単体では動作しません。これはSAMPLE1からSAMPLE2のプログラムのちがいだけを記述しています。ようするにSAMPLE1にSAMPLE3をマージすれば、SAMPLE2ができあがるということです。プログラムの追加部分を確認しましょう。

さて次回は、ガラッと内容をかえてサークシリーズのオープニングなどで使われている手法のどれかを説明しようかと考えています。リクエストがあればお便りください。また、ハイテクコーナーあてにも教えてほしいテーマがあったらくださいね。もちろん一番忘れてはいけない紙芝居の投稿。こちらも心からお待ちしてま～す。

【圭一愛の物語3本パック】

圭一愛の物語Ⅰ

92年5月号掲載作品

ドラゴン討伐をする第1話。優子ちゃんの危険に身をていして立ち向かう圭一

圭一愛の物語Ⅱ

92年5月号掲載作品

ダンジョンには危険がいっぱい。圭一の友情エピソードをえがいた第2話

圭一愛の物語Ⅲ

92年7月号掲載作品

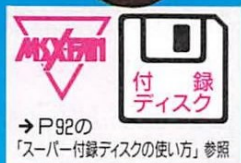
この世に平和がおとずれ、夢だった学園生活。愛のバレンタインで圭一ドキドキ♡

今回のPOINT/本当の意味での「紙芝居」に挑戦してみよう

いままでの授業では、「コンピューターアニメーション」を中心に教えてきましたが、今回は「紙芝居」を教えたいと思います。アニメ処理をしない1枚絵でも、絵と字によって、楽しい作品を作ることができるようになりますよ。

→P92の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

講師プロフィール
株式会社マイクロキャビン・企画開発課主任。MSX2の『ザーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

グラフィック画面に文字を出そう

マスターアップだあァァァァァ!! と右往左往している中津です。えーと、CMでーす。3DOで6月ごろ新作を出します。チェックよろしく/

さて今回は、ちょっと予定を変更して紙芝居入門みたいな感じでお届けします。いままでの授業ではコンピュータでアニメをさせるという事をメインで進めてきましたが、今回は本当の意味での紙芝居、すなわち絵と文字を同時に出すというものに挑戦しましょう。そのためいろいろと説明することが多くて、ハイテクコーナーはお休みです。ごめんなさい。

では、授業に入ります。まず「グラフィック画面に文字を出す」ということについてから説明していきましょう。

BASICのリストを打ちこむSCREEN0や1の場合とちがって、紙芝居でよく使うSCREEN5はグラフィック画面です。このグラフィック画面になっているときは、PRINT命令で文字を出そうとしても表示してくれません。では、どうすればよいのでしょうか?

答えはOPEN命令を使えば解決します。ふつうOPEN命令はフロッピーディスクなどに、データを書きこんだり読みこんだりするために使いますが、さすがMSXのBASIC、ふつうじゃない使い方もサポートしていたりじます。書式は以下の通りです。

```
OPEN "grp:" FOR OUTPUT AS #1
```

こうすると、#1というファイル番号でグラフィック画面に文字を出すことが可能となります。ではじっさいに出力をしてみましょう。文字の出力にはBASICと同じPRINT命令を使います。ただし、書式がちょっとかわります。

```
PRINT #1,"テスト"
```

これで、グラフィック画面にテストという文字が出力されます。かんたんにいえば#1をつけることで、" "でくくられた文字をグラフィック画面上に出力することができるということです。原理は下のイラストを参照してくださいね。

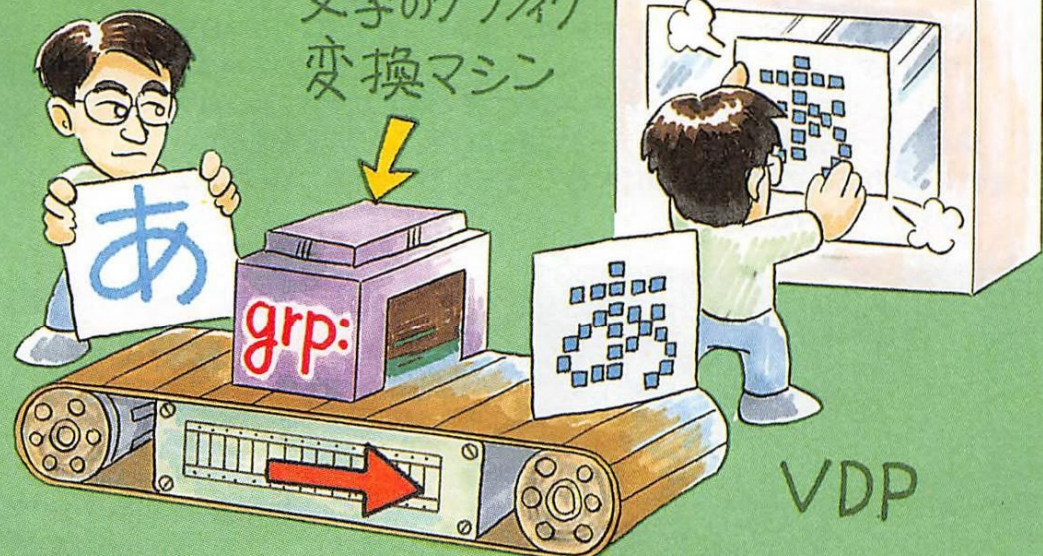
さて、もうひとつ。画面内の好きな位置に文字を出したいときはどうしたらよいのでしょうか。これがSCREEN0や1ならLOCATE命令を使いますが、グラフィック画面ではこの命令は使えません。そこでPSET命令を使います。

このPSET命令は本来グラフィック画面に点(ドット)を表示するためのものですが、こういった任意の位置に文字を出すという目的にも使用できます。書式は以下の通りになります。

```
PSET(X座標,Y座標)
```

文字を表示させたい位置をグラフィック座標で指定する以外は、LOCATE命令と似ています。これだけで好きな位置に文字を出せるようにできるのです。

⬆本当の意味での「紙芝居」。それは絵と字で作っていく物語。やってみるとけっこうかんたんだから、みんなも挑戦してね

グラフィック画面に文字を表示するしくみ

グラフィック画面上ではテキスト文字を表示できません。OPEN命令を使って「あ」という文字を表示したとしても、それはドットで構成された「あ」に変換されているのです。

データを使ってすっきり整頓

数値データをすっきり整理するにはどうしたらいいのでしょう。最近の授業では、プログラムでいちいち書くより、ハコを用意してかんたんにまとめています。むずかしくてわからない人は、プログラムをいちから書いていくのもいいでしょう。でも、ちょっとなれた人なら、ぜひハコを使ってみてください。プログラム自体がぐっと短くシンプルになりますよ。

今回のサンプルプログラムでは、新たに配列というものを使っています。これを使うと、いままで以上にうまくデータが整理できるので、プログラムをさらにシンプルにしてバグを少なくしてくれます。見やすいのでデバッグ時にも便利です。

配列とは、ハコの特別な使い方です。いままでは、1個のハコの中に1個の数値を入れて使ってきました。でも、この配列を使うことによって、1個のハコの中に複数のデータを入れることができるようになるのです。考え方として、1個のハコの中を複数に区切ると思ってもらえばいいです。たとえば、ハコをマンションと考えれば、区切ったものはそれぞれの部屋ということにたとえられますね。

さて、それではじっさいにやってみましょう。ハコを区切るのには、DIM文を使います。書き方は以下の通りです。

```
DIM GX(4), GY(4)
```

GXやGYというのはハコの名前、その後ろにある(4)というのは、それぞれのハコの中に区切る場所の数を示しています。この場合、ハコGXの中に5つ(0から数えて4は5つ目なので)、ハコGYの中に5つの場所を作ったことになります。

それでは中身を入れてみましょう。これに中身を入れるには、

```
GX(0)=10
```

というように書きます。(0)というのはハコの区切り番号です。最初DIM文でハコGXの中を区切って5つの場所を作りましたね。この5つの場所はそれぞれGX(0)〜GX(4)といったようにナンバーがふられて分けられているのです。

ですからこの「GX(0)=10」の場合、ハコGXの0番目の場所に数値10が入るということを示します。値を入れた配列のハコは、ふつうのハコと同様に計算に使うことができます。いままで使ってきたハコの使い方と同じというわけです。

さて、こんな便利に使える配列にも問題がひとつ。もし、データが多くなってきた場合どうなるでしょう。人によってはいいという人もいそうですが、たくさんの配列の番号と、入れる数値をいちいち書きならべるのはたいへんな作業です。もし、100個もあったら気が遠くなってしまいますしね。私はいやです。

それでは、もうちょっと楽をする方法はないのでしょうか?

実はあります。それには、READ文とDATA文を使うと楽にすることができます。

DATA文は、数値データなどを「,」で区切って表形式に書きならべることができます。数値データはデータとしてあとからまとめて書いておくのです。こうすることでプログラム全体を見やすくし、打ち込みのミスをへらしたり、あとからのバグ修正も楽になるわけです。

DATA文の書き方は以下の通りです。

```
DATA 10, 9
```

これを配列に入れるには、READ文を使います。READ文は、DATA文の定義したデータを行番号の若い順から読みこんできてハコや配列に入れてくれる働きがあります。

READ文の書き方は以下の通りになります。

```
READ GX(0), GY(0)
```

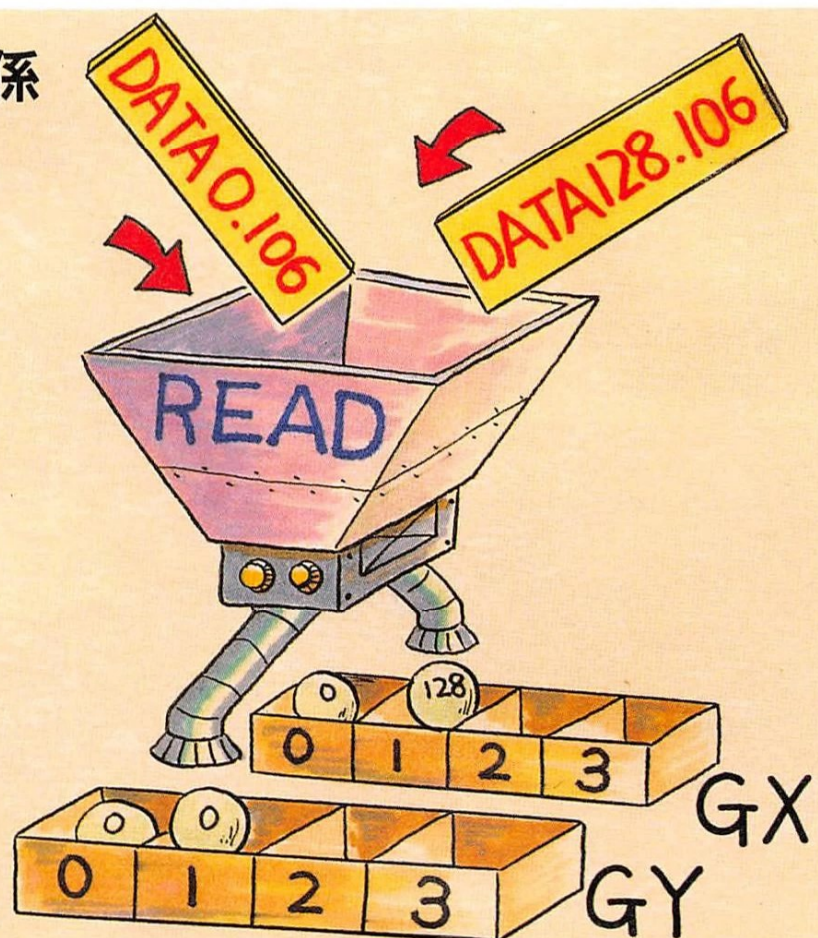
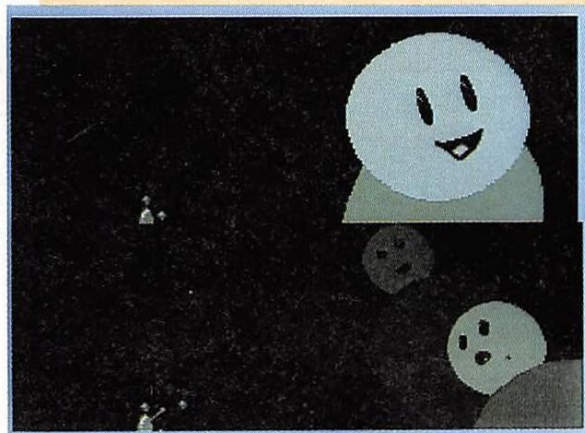
こうすれば、さきほど定義したDATA文の数値を読みにいき、GXの0番目のハコに10が、GYの0番目のハコに9が入るというわけです。

この例では、2個だけですがもっとDATA文がふえたら、FOR〜NEXT文を使って( )の中をFORのハコ(変数)にしてやればシンプルにできるのです。今回は基本ということで、そのことはまたいつかやりましょう。

それでは、DIM文、READ文、DATA文、を忘れないうちに、今回のサンプルプログラムの説明に入ります。

今回はキャビンオリジナルの紙芝居です。

READ文、DATA文の親密な関係

初心者の方は、DATA文でまとめて書いておいた数値をREAD文がどのように読みにいくのかがわからないといいますが、これはじつにかんたんです。例えば、DATA 1番目の数値, 2番目の数値というように数値をならべておけば、READ文のほうも、READ 1番目の数値, 2番目の数値をそれぞれ読みにいきます。こうして、READ文とDATA文を使って数値だけをまとめておくことにより、いちいちハコを用意していく手間をはぶくことができるのです。今回のサンプルプログラムでは、CGを転送する座標データにREAD文、DATA文を使っています。それぞれの座標をハコに入れて、READ文で読みにいっているというわけですね。

プログラムを組んでみよう

5〜10　初期化。
20　ＤＩＭ文でGX、GYの配列を5つの場所に区切っています(今回使うのは4つです)。
30　グラフィック画面に＃１のファイル番号でOPENしています。これで、グラフィック画面上(この場合はSCREEN 5)で文字を出す準備が完了したわけです。
40〜60　DATA文で定義されたデータを配列GX、GYに順番に収める。
700〜740　紙芝居用のグラフィックデータ座標です。DATA文の中の１個目がX座標、2個目がY座標にあたります。
100〜190　紙芝居のグラフィックデータをPAGE１にロードしています。
200〜240　タイトルの表示。表示後､60分の300秒ウエイトしています。
300〜450　１〜5コマ目の絵とセリフの表示制御。こまかい処理は､行800からのサブルーチンが行なっています。ハコＩは絵の番号。ハコC$は、出力するセリフです。
500〜550　エンドタイトルの表示。
560〜570　キー入力を待って、プログラム終了。
800　絵とセリフの表示サブルーチン。
810　ハコＩの示す番号のグラフィックデータの座標を配列GX、GYからX、Yに取り出しています。
820　文字を表示する部分をLINE文でクリアします。BFは塗りつぶしの意味です。2点をつなぐ対角線を持つ四角形の範囲を、色コード０で塗りつぶしています。
830　グラフィックをハコX、Yの示すPAGE１の座標から転送して表示。
840〜850　PSET座標へ、PRINT　#１を使ってC$に入っているセリフを表示しています。
860　60分の200秒ウエイト。
890　リターン。

以上が今回のサンプルプログラムです。むずかしそうに見えるかも知れませんが、作ってみると案外かんたんですよ。じっくり見合わせてくださいね。

今回の紙芝居は、絵と字を出すということ。いままでやってきたアニメが作れなかった人でも、１枚の絵と字を組み合わせれば、かんたんな「紙芝居」はできます。ポイントは、今回やったグラフィック画面に文字表示するOPEN文(P15)、そしていままでやってきた絵の転送のCOPY文。これだけでも4コママンガみたいなものを作ることもできますね。

そこで、最近投稿がへって私は悲しいぞということで、今回のプログラムとデータを使って、君も紙芝居に挑戦してみないかキャンペーンだ／　今回のサンプルプログラムのメッセージや絵のデータをかえて、投稿してください。セリフやコマ並びを変更しただけでもOK。

もちろん、できる人はオリジナルの紙芝居を作ってきてほしいな。キミの投稿をおまちしていまーす。　　　(中津)

SAMPLE. BAS

```
5 DEFINT A-Z
10 SCREEN 5:CLS
20 DIM GX(4),GY(4)
30 OPEN "grp:" FOR OUTPUT AS #1
40 FOR I=0 TO 3
50 READ GX(I),GY(I)
60 NEXT
100 SET PAGE 0,1
110 BLOAD"a.ge5",S
120 COLOR=RESTORE
180 TIME=0
190 SET PAGE 0,0
200 'ﾒｲﾝﾌﾟﾛｸﾞﾗﾑ
210 CLS
220 PSET( 64 ,8*10 )
230 PRINT #1,"◆◆ ほし に ねかﾞいを ◆◆"
240 IF TIME<300 THEN 240 ELSE TIME=0
300 'no.1
310 I=0:C$="あっ!なかﾞれほﾞしたﾞ!"
320 GOSUB 800
330 'no.2
340 I=3:C$="とﾞんな ねかﾞいを かけたんたﾞい"
350 GOSUB 800
360 'no.3
370 I=1:C$="ううん｡まにあわなかったの!"
380 GOSUB 800
400 'no.4
410 I=0:C$="あっ!またたﾞ! えっと!なんにしようかな?!"
420 GOSUB 800
430 'no.5
440 I=2:C$="はよせんか!"
450 GOSUB 800
500 CLS
510 PSET( 8*2,8*10 )
520 PRINT #1,"    presented by MICRO CABI
N"
530 PRINT #1," "
540 PRINT #1,"          CG          ﾅｶﾞｲ ﾉﾘﾋﾛ"
550 PRINT #1,"          program   ﾅｶﾂ  ﾔｽﾋｺ"
560 A$=INPUT$(1)
570 END
700 'さﾞひょうﾃﾞｰﾀ
710 DATA 0,0
720 DATA 128,0
730 DATA 0,106
740 DATA 128,106
800 'ひょうしﾞｻﾌﾞﾙｰﾁﾝ
810 X=GX(I):Y=GY(I)
820 LINE (0,8*20)-(255,199),0,BF
830 COPY (X,Y)-(X+127,Y+105),1 TO (64,50
),0
840 PSET( 8*6,8*20 )
850 PRINT #1,C$
860 IF TIME<200 THEN 860 ELSE TIME=0
890 RETURN
```

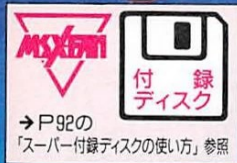
AFTER SCHOOL
紙芝居倶楽部

今回の付録ディスクでは、またしても投稿がなかったのと、付録ディスクの容量のかねあいもあってサンプルプログラムのみとなってしまいました。やっぱり投稿がないと、紙芝居倶楽部のスペースも小さくなってしまいましたね。私は悲しいぞ、しくしく。

しかし／　この原稿を書いている時点で投稿がきました／　今回は残念ながら間に合いませんでしたが、次回こそはまた大きく発表できそうです。わくわく。

さて、それでは今回のサンプルプログラムについて説明しましょう。今回はいままでのアニメとちがい、いかにも紙芝居的なお話をキャビンからお贈りします。制作者は私、中津と、グラフィッカーの永井先生(パチパチ)との共同制作です。ぜひ楽しんでください。

グラフィック画面に文字を出すということで、たいへんシンプルなものになりました。これなら初心者の方でも、自分で紙芝居を作れると思います。みなさん、ぜひ挑戦してくださいね。

↑流れ星に願いをかける親子の、感動(?)の物語が今回のサンプルです

マイクロキャビン
中津泰彦

今回のPOINT／紙芝居にとって大切なDATA文

今回は、前回の授業でやったOPEN、PRINT、PSET、DATA文を、さらに奥をついて説明しましょう。それから久しぶりの読者投稿もありましたので、こちらもしっかり紹介して、全4ページでお送りしまーす。

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の『サーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

電光掲示板を作ってみよう

さあ、始まりました紙芝居&動画教室。いつものCMは最後のページにありますので、さっそく授業に入りましょう／

今回の授業は、前回やった「紙芝居」の続きです。まずはグラフィック画面に文字を表示する方法から。

「SAMPLE1．BAS」を実行すると、右から左へ、MSXのテキスト文字が流れていきます。でも、テキスト文字はそのままでは流せないので、この文字はグラフィックに変換された文字であるというわけです。

文字をグラフィック変換するには前回も説明した、OPEN、PRINT、PSETの命令を組み合わせて使用します。これを使うことにより、SCREEN5などのグラフィック画面に文字を出すことが可能となります。変換された文字はドット単位であつかえるので、これをいかしているわけですね。

それでは前回のおさらいも含めて説明していきましょう。

OPEN命令はグラフィック文字を使う機能を起動します。この機能は出力先のファイル名の部分に"GRP："を指定することで有効になります。そしてそのとき、＃1などのファイル番号を指定して定義する必要があるので注意しましょう。このファイル番号をとおして画面に文字を送るからです。ただし、通常使えるファイル番号は＃1だけになっています。

PRINT命令では、指定されたファイル番号を使って文字をグラフィックに変換して画面に出力します。PSET命令は、PRINT命令で出力されるグラフィック文字の座標をセットします。

プログラムの仕組みとしては、裏画面に文字のバッファを用意して、このバッファの一番後ろの部分に1文字ずつグラフィックに変換して送りこんでやります。送りこまれたグラフィック文字は、COPY命令を使って1ドットずつ左に転送することでシフトしてやります。同時に表画面の掲示板に後半8ドットを除いた部分をCOPY命令を使って表示します。これを8ドットシフトするまでくり返したあと、次の1文字を送りこむのです。これで、見た目には右から左に文字が流れてきます。

20　SCREEN5に設定
30　グラフィック画面にファイル番号＃1で文字を出力可能にします
40　LINE文をつかって電光掲示板の枠を作ります
50〜60　(64，64)の座標にタイトルを表示します
80　ハコSR$に電光掲示板に表示する文字列を定義
90　ページ0を表画面、ページ1を裏画面に。文字は、裏画面のバッファに出力されます
100　文字列の長さのぶんだけループします
110　8ドットぶんのループ
120　COPY命令を使って、バッファから表画面に表示
130　COPY命令を使って、左へ1ドットだけバッファをシフトします
140　ウエイトループです
170〜180　ハコSR$の中から1文字だけMID$関数を使って取り出す(この場合は、1番目の1文字)。その文字をPRINT命令で、裏画面のバッファに出力しています

↑右から左へスムーズに文字が流れます。ふつうのテキスト文字だったらこうはいきません

SAMPLE1. BAS

```
10 'さんぷる 1:でんこうけいじばん
20 SCREEN 5:CLS
30 OPEN "grp:" FOR OUTPUT AS #1
40 LINE (63,94)-(184,104),15,B
50 PSET(64,64)
60 PRINT #1,"MSXでんこう けいじばん"
80 SR$="えー つまんないだじゃれを ひとつ。              ねこ
が       ねこんだ          ねころんだ!
 おそまつさまでしたー!                     "
90 SET PAGE 0,1:CLS
100 FOR I=1 TO LEN(SR$)
110 FOR S=0 TO 7
120 COPY (0,0)-(119,7),1 TO (64,96),0
130 COPY (1,0)-(135,7),1 TO (0,0),1
140 FOR W=0 TO 10:NEXT
150 NEXT
170 PSET(120,0)
180 PRINT #1,MID$(SR$,I,1)
190 NEXT
```

大作を作るならDATA文を有効活用せよ！

つづいて、DATA文の使い方の説明をしていきましょう。付録ディスクにサンプルプログラムの「SAMPLE2．BAS」を収録してありますので、そちらをご覧ください。それでは説明に入ります。

DATA文はいろいろなデータを整理するのに向いています。これは、関連するデータを表形式にならべらることができるため、データの打ちこみや修正、追加がかんたんになるというメリットがあります。紙芝居や動画を作るとき、そのプログラム作成中にいちばん多い作業といえばデータの修正や追加にほかなりません。DATA文を使いこなせれば、あなたの作業効率は何倍もアップすることでしょう。ぜひ覚えましょう。

DATA文を使うのに重要なBASICの命令は、前回の授業でやった、DIM文とREAD～DATA文です。

まずはDIM文から説明していきましょう。

DIM文は、配列変数を定義するものです。配列変数とは、ひとつの大きなハコを区切ってたくさんのデータを収められるようにするものです。このハコひとつでたくさんのデータが覚えられるので、物忘れの多い私でもハコの名前をおぼえていられるんです（／）。データの数だけハコを用意したら、どのデータがどのハコにあったかさすがに覚えていられませんからね。

前回の説明でいい忘れましたが、この配列変数を定義するにあたり、ひとつ注意があります。たとえば、

DIM A(2)

と宣言して、Aのハコの中を0から2まで区切って部屋を3つ定義してあげたとしましょう。このハコAの0番目の部屋に数値5を入れたいときは、

A(Ø)=5

とすればよいわけですが、この配列変数で定義したハコAと、ふつうのハコAとはまったくの別モノになっているということを覚えておいてください。

具体的にいえばこの場合、ひとつのハコAと、配列変数として定義された、たくさんの部屋を持つハコAがあるということです。この2つのハコAはそれぞれまったくちがうハコなので、まちがえないようにしましょう。

つづいてREAD～DATA文の説明に入ります。

READ文は、DATA文で定義されたデータを読み出してDIM文で定義した配列などに入れてあげることができます。

今回のサンプルプログラムでは、一定の大きさを持たないグラフィックデータと、グラフィックを動かすためのアニメデータを、大きく2つのDATA文にわけて処理してみました。

まずひとつ目のDATA文。これはグラフィックのデータを定義しています。その中身は、裏画面に置かれたグラフィックデータの座標と、その絵のサイズです。

これらを配列に入れることで、実際にあつかうときに番号で処理できるようになります。たとえば、1枚目のグラフィックデータを表画面の座標(10，10)に表示しろということもかんたんに書けるようになりますし、何よりもサブルーチンにするときに便利になるのが大きなポイントとなりますね。

そしてもうひとつのDATA文。ここではさきほど定義したグラフィックのDATA文とは別に、動きのデータを定義します。データの内容は、表示先のX，Y座標と、表示するグラフィックの順番です。

今回のサンプルプログラムでは、大きく2つに分けたDATA文から、1データずつ引き出して、2つのデータを合わせ、最終的にCOPY命令のパラメータとして使用しています。

次ページに掲載したサンプルプログラムを見ていただけるとわかると思いますが、もし配列やDATA文を使わなければ、かなり複雑なプログラムになることは予想できるのではないでしょうか？

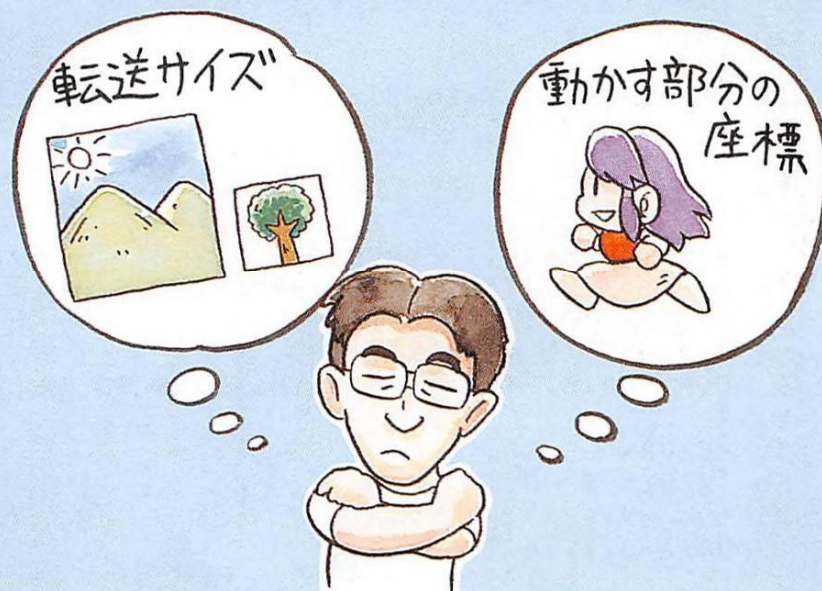
プロのプログラムでは、ほとんどのデータをこのやり方の応用で処理しています。ですから、このDATA文がいかに重要で基本的な技術かわかっていただけると思います。

大きなプログラムは数値が迷いがち

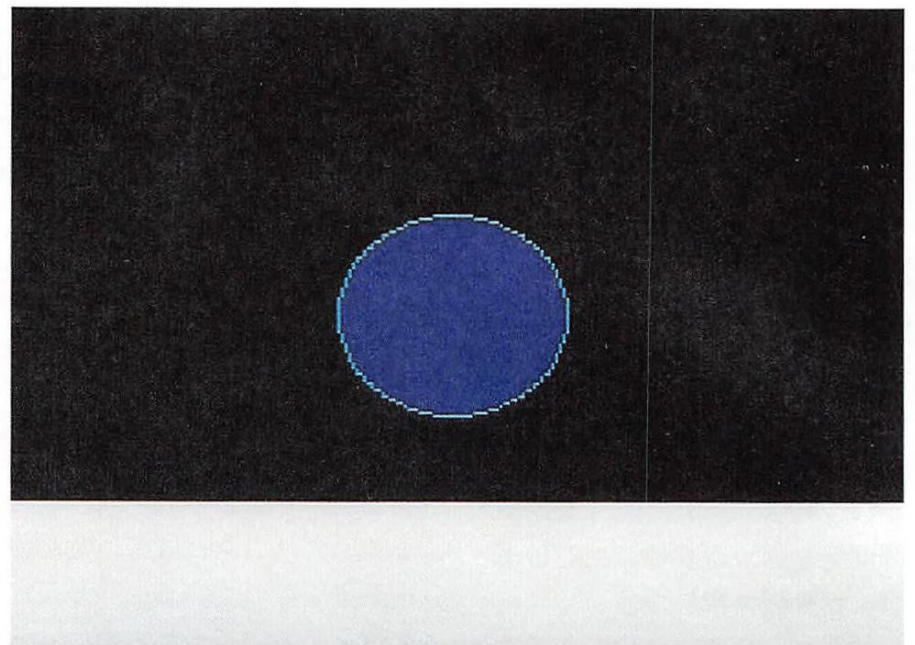
あれ！？転送するサイズって
いくつだっけ？　座標は…？

DATA文であとから数値データだけをまとめて書いておけば、プログラムがとても短く、見やすく、まとめやすくなります。転送のたびにCOPY命令をいちいち使っていたのでは、大きなプログラムを扱うとき、数値だらけで頭の中が混乱しますよ。初心者の方もDATA文はぜひ覚えてくださいね。

いろいろ使えるDATA文

DATA文は転送サイズや動きの座標など、分けて使用すると便利です。

サンプルプログラムで頭の中を整理しよう

では、プログラムの説明です。
20〜70　初期化。SCREEN5の画面モードにして、"MMM.GE5"のグラフィックデータをページ1に読みこんでいます
80〜90　グラフィックの定義データの配列変数を宣言します。定義する部屋の数は8です
95　移動定義データの配列を宣言します
100〜120　行510からのDATA文からグラフィックの定義データをREAD文で読みこみます。定義データは7つです
130〜150　行610からのDATA文から移動定義データをREAD文で読みこみます。終了条件は、読みこんだデータのX座標が−1になったときです
170〜180　表ページに地面をLINE文で作っています
210　移動定義データを先頭から実行します
220　データのX座標が−1なら、もう一度行210に戻ってはじめからくりかえします
230　行300からのサブルーチンで実際に表示します
240〜250　次のデータに移ります。行220へ
310　わかりやすくするために、別のハコにデータを取り出す。ハコNはグラフィックデータの番号。Xは表示するX座標、Yは表示するY座標です
320　COPY命令を使って表画面にグラフィックを転送します。転送元の座標は、グラフィックの定義データを使用
330　ウエイトループ
340　ちょっと卑怯な技を使用。COPY命令を使って表画面のすみっこから真っ黒なグラフィックを持ってきて、上に重ねてグラフィックを消しています。これはキタナイのであんまり真似しないこと
350　もとの処理にもどります
510〜570　グラフィックの定義データ。左からX座標、Y座標、Xサイズ、Yサイズ
610〜900　移動定義データ。左から表示先X座標、表示先Y座標、表示するグラフィックデータの番号(行510からのDATA文で定義されたもの)

以上が、プログラムの説明です。次回では、この2つを応用して、紙芝居で何ができるかに迫ってみたいと思います。

ボールがぽんぽんと跳ねます。プログラムでは移動と座標のデータをわけているのです

SAMPLE2. BAS

```
10 'さんぷる 2:dataぶん
20 SCREEN 5
30 COLOR 15,1,Ø
40 CLS
50 SET PAGE Ø,1
60 BLOAD"mmm.ge5",S
70 COLOR=RESTORE
80 NN=7
90 DIM X(NN),Y(NN),XS(NN),YS(NN)
95 DIM XP(4Ø),YP(4Ø),NO(4Ø)
100 FOR I=Ø TO NN-1
110 READ X(I),Y(I),XS(I),YS(I)
120 NEXT
130 I=Ø
140 READ XP(I),YP(I),NO(I)
150 IF -1<> XP(I) THEN I=I+1:GOTO 14Ø
170 SET PAGE Ø,Ø
180 LINE (Ø,154)-(255,2ØØ),14,BF
200 'ひょうじ メイン
210 I=Ø
220 IF -1=XP(I) THEN 21Ø
230 GOSUB 3ØØ
240 I=I+1
250 GOTO 22Ø
300 'ひょうじ サブ
310 N=NO(I):X=XP(I):Y=YP(I)
320 COPY (X(N),Y(N))-(X(N)+XS(N)-1,Y(N)+
YS(N)-1),1 TO (X,Y),Ø,TPSET
330 FOR W=Ø TO 4Ø:NEXT
340 COPY (Ø,Ø)-(XS(N)-1,YS(N)-1),Ø TO (X
,Y),Ø
350 RETURN
500 'ざひょう データ
505 '      x,y,xサイズ,Yサイズ
510 DATA Ø,Ø,16,16
520 DATA 16,Ø,32,32
530 DATA 48,Ø,48,48
540 DATA 96,Ø,64,64
550 DATA 16Ø,Ø,64,48
560 DATA 16Ø,48,64,32
570 DATA 16Ø,8Ø,64,16
600 'いどう データ
605 '      X,Y,ヒョウジデータno
610 DATA 9Ø,ØØ,3
620 DATA 9Ø,Ø2,3
630 DATA 9Ø,Ø4,3
640 DATA 9Ø,Ø8,3
650 DATA 9Ø,16,3
660 DATA 9Ø,32,3
670 DATA 9Ø,64,3
680 DATA 9Ø,9Ø,3
700 DATA 9Ø,1Ø6,4
710 DATA 9Ø,122,5
720 DATA 9Ø,138,6
730 DATA 9Ø,122,5
740 DATA 9Ø,1Ø6,4
750 DATA 9Ø,9Ø,3
760 DATA 9Ø,64,3
770 DATA 9Ø,32,3
780 DATA 9Ø,3Ø,3
790 DATA 9Ø,16,3
800 DATA 9Ø,Ø4,3
810 DATA 9Ø,Ø2,3
820 DATA 98,ØØ,2
830 DATA 1Ø6,ØØ,1
840 DATA 114,ØØ,Ø
850 DATA 114,ØØ,Ø
860 DATA 1Ø6,ØØ,1
870 DATA 98,ØØ,2
900 DATA -1,-1,-1
```

AFTER SCHOOL

紙芝居倶楽部 ～投稿作品より～

【ミリーのぼうけん】 山口県／REDS(21歳) SCREEN5・4分の1画面サイズ

今回は投稿が４作品もきました。ありがとうございます。では、作品を紹介しましょう。まず、いちばんすばらしかったのがこの『ミリーのぼうけん』。文字表示もない、音もない、いわゆるサイレンスアニメですが、途中でグラフィックデータを１回読みにいくほどの大作に仕上がっています。この作品は絵をスプライトにすることで、高速な表示とおどろくべき動画枚数を可能にした技術と努力の作品です。この作品のおもしろいところは、ただ動画を再生できるだけでなく、巻き戻しや逆再生などができる点でしょう。画面下に表示されているのがその操作パネルです。みなさんも遊んでみましょう。それからなによりも、全部で192枚の動画を描いたことに頭が下がります。

戦うミリー

にげるスライム

【文鳥】 埼玉県／12＋47(12歳) SCREEN5・4分の1画面サイズ

つづいてはこの『文鳥』。文字どおり文鳥が出てきて踊ります。鳥が実際よくやるキョロキョロとした動きは、よく観察していますね。とても似ています。また、この正面を向いた文鳥の顔がかわいいこと！ 中津は気に入りました。そしてこの作者はなんと12歳。すごい！ お父さんと作ったとはいえ、将来が期待できます。この作品は、今までのこの教室で紹介してきたテクニックを応用し、文鳥に対する愛のあふれる作品となってます。プログラムを見ていただくとわかりますが、ＣＯＰＹ命令を使ってひとつひとつ作られています。最初はこのやり方で正解です。今の授業でやっているＤＡＴＡ文などは、それより１歩ステップアップした形なので、初心者の方はＣＯＰＹ命令を多用するのがよい方法です。今後はより高度なテクニックを覚えて、より楽しい作品作りに挑戦してください。

つんつん

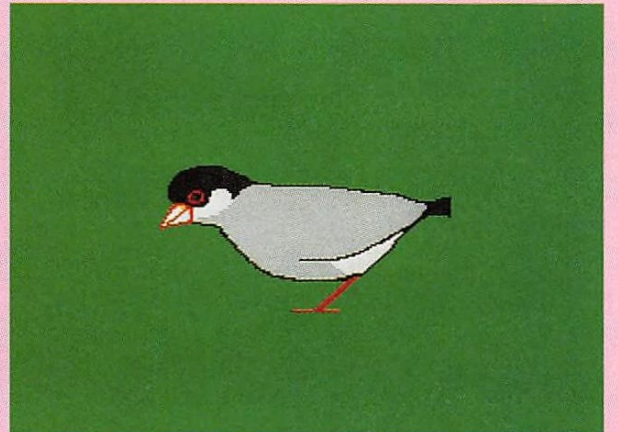
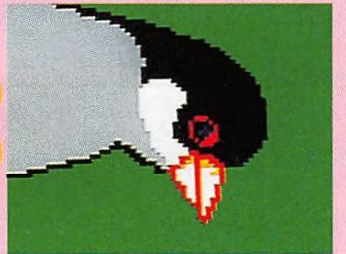
かわいい！

おしくもボツになったけど……

こちらは、残り２作品。今回はおしくもボツになってしまったものですが、中津から作者にアドバイスをば。

まずこの「炎」ですが、テーマ性があり、個性あふれる作品です。ただ、舞台背景がわかりにくいなどの「人に見せる」という部分での練りこみが不足しています。友達とかに見せて、その人がどう思うかについての研究をしてみてください。人に見せると、意外といい答えが返ってくるものですよ。

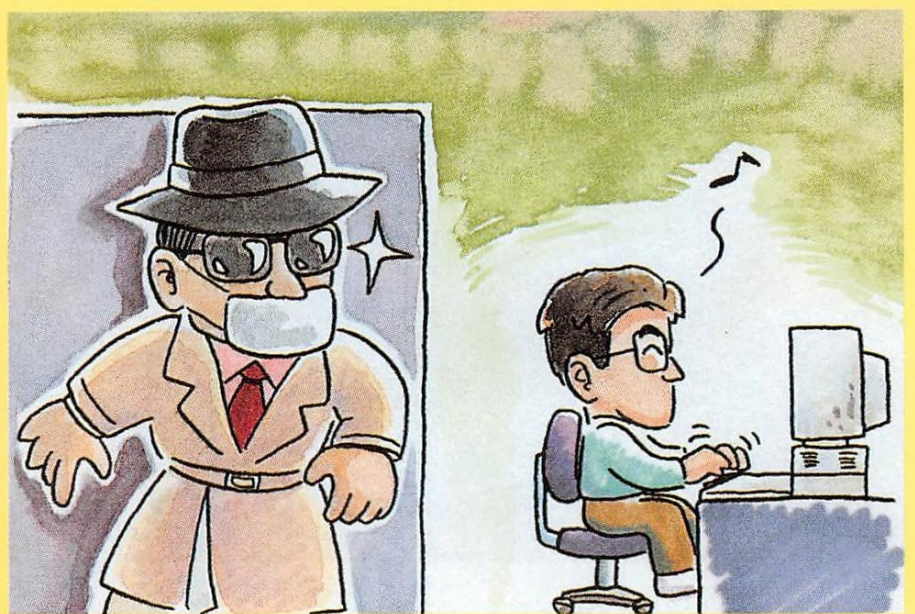
【家出王子「炎」の夜】
埼玉県／天使単色(17歳)

↑文章による舞台背景より、見てわかる舞台背景を表現するように心がけよう

こちらの作品は、近未来の戦闘を描いたミリタリー色あふれる作品。この作品のおしい点は、描いたものを見せようとしすぎて、テンポがスローになっているところです。もっとアップテンポで進めないと戦闘シミュレーションの雰囲気が生きてきませんし、見ているほうもたいくつです。あと、オチが弱いですね。もっとシリアスな結末でもよかったのでは？ それから、ここまで作るのなら、ゲームとして作ってもよかったかも？

【V. R. BATTLE in 2008】
北海道／陸井ドン//(17歳)

↑紙芝居にするより、この表示を生かしたゲームにするほうがおもしろかったかな？

中津先生とホームルーム

ここはマイクロキャビンの情報コーナーらしい。そしてそれをひそかにさぐる私は、探偵のギャブ・モブラードだ。以後よろしく。今日は某機関の要請で、最近の中津の行動を調べたので聞いていただこう。最近の中津は、セ〇のサタ〇ンという機械でＲＰＧを作っているらしい(雑誌にはすでに発表済みだそうである)。なんでも３Ｄポリゴンをフィールド表示に応用した新しい形のＳＲＰＧだということだが、それ以上は調べることはできなかった。仕事の内容は、ゲームシステムのデザインとメインのプログラムということで、目も回るいそがしさだと周囲の人にもらしており、さらにあの『幻影都市』の核となった人物の三人のうち二人まで参加している作品なので、ストーリーには期待してほしいと大きい口をたたいているとのことである。以上報告を終わる。次回では、さらにこの中津なる人物について追っていこうと思う(ザーッ)。

紙芝居&動画教室

今月のPOINT／DATA文で紙芝居をコントロールする

今回は、前々回、前回の授業でやった、紙芝居と電光掲示板を組み合わせて、さらに自然なビジュアルシーンを目指します。とくに紙芝居をコントロールするDATA文に注目してください。

MSXFAN 付録ディスク

→P85の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の『サーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

紙芝居に電光掲示板を組み合わせる

こんにちは。3DOのゲームソフトの開発が終って間もないのにすでに、めちゃくちゃ忙しいというハイな状態になっている中津です。

今回の講義に入る前にまず、おさらいをしておきます。

前々回は絵と文字だけの紙芝居を題材に、グラフィックに文字を表示するしくみ、READ～DATA文でCGデータの座標を管理するテクニックを覚えました。

前回は電光掲示板を題材に、一歩進んだ文字表示のテクニックとDATA文の便利さを紹介しました。

今回の紙芝居教室は、これらのテクニックをもとに、前々回の紙芝居をバージョンアップしてみましょう。前回取り上げた1文字ずつグラフィック文字を操作するテクニックとDATA文を使ってプログラムをコンパクトにまとめ上げることを行います。

今回は、CGデータを含むほとんどのデータと紙芝居のコントロールをDATA文の形で記述しています。このため、データの形として非常にチェックしやすく、1画面に納まる見通しの良いデータとなりました。

DATA文で記述されたデータは、2つのブロックに分けて記述してあります。

1つはCGデータの場所を記述したもので、格納してある座標X、Yと画面のページ番号を定義します。

このように処理しデータを配列に読み込むと、各CGデータを番号で管理できるようになるので、物覚えの悪い人間としては非常に助かるのです。

もう1つのデータは、紙芝居の流れをコントロールするデータです。紙芝居のシナリオといってもいいでしょう。このデータの内訳は、表示するCGデータの番号、ウエイト(次の文字を表示するまでの待ち時間のこと)、表示するセリフという順番になっています。

紙芝居では、ウエイトによるセリフのスピードで感情や時間の流れを表現することも重要なテクニックとなりますので、今回は簡単ですが採用してみました。サークシリーズのビジュアルシーンのセリフも注意してみると参考になるでしょう。

今回のように、DATA文はいろいろな形で、いろいろなデータを扱えます。ただのセリフにしても、表示開始位置、文字の色、大きさ、ウエイト、文字間隔などなど、さまざまな要素が考えられ、これらはやはりデータにしておくのです。

また、プログラムのなかで使っている変数がプログラムの処理に本当に必要なものなのか、それともたんなるデータなのかを見極めることがとても重要になってきます。そして、データはDATA文にまとめてしまうと、見やすくて拡張も簡単なプログラムになります。

「星に願いを・パートII」です。前作の約2倍の長さで、タップリ楽しめます。なんだかとんでもないオチが……

パラメータはすべてDATA文に

CGデータの番号もウエイトもセリフ本体もすべてDATA文のデータになっています。これを読み出すREAD文とその変数を工夫すれば、どんな処理にもDATA文を使うことができますよ。

DATA文のデータ構造を工夫して

10〜20　初期化。画面モードをスクリーン5にして、クリア
30〜35　ＤＩＭ文で配列を定義します。GX、GY、PGはCGデータを管理するものでそれぞれX、Yとページを格納します。DT、WT、C$は紙芝居のコントロールをするもので、CGデータ番号、ウエイト、セリフが入ります。C$は特別で、文字列を格納する配列です
40　グラフィック画面に＃１のファイル番号で、グラフィック文字を出力可能にします
45〜70　配列GX、GY、PGに行700からのCG管理データを読み込んで初期化します
80〜110　配列ＤＴ、ＷＴ、C$に行800からの紙芝居データ（シナリオ）を読み込みます。データに－１がくれば終了です
120〜140　ページ１にCGデータ１を読み込み、カラーパレットを設定します
150〜160　ページ2にCGデータ2を読み込みます
170　グラフィックの表示と処理ページを０に設定します。グラフィック文字を扱う場合、この設定を忘れないようにしましょう。そうしないと、バグの原因がわからず悩むことになります
200　ここからはメインのプログラムです
210　取り出すデータの番号Ｄを０にします
220〜230　タイトルであるCGデータ４をサブルーチン（行500からのもの）で表示、行230はウエイトです
240　Ｄの示すデータを取り出して、GN、S$、Wに格納します
250　GNが－１ならば、終了なので行300へ
260　表示サブルーチン（行500）へ
270　Ｄを＋１して次のデータを取り出す準備
280　行240に戻る
300　画面をクリア
310〜320　グラフィック文字で、おしまいの表示
390〜400　なにか押されたら終了
500　表示サブルーチン
510　GNの示すCGデータのX、Y座標を取り出します
520　文字を表示する部分をＬＩＮＥ文で塗り潰す。要するに、前に書いた文字を消します
530　CGデータを表示画面に転送します。CGデータは配列X、Yの座標と配列PGの示すページ番号の場所が転送元です
540　文字列の表示開始座標を設定します
550　１文字ずつ取り出して表示するため、ＦＯＲループになっています。関数ＬＥＮは、文字列の長さを返す関数です
560　Ｉ番目の１文字を取り出して、グラフィック画面に表示します
570　ウエイト値のW／60秒だけウエイトします
590〜595　50／60秒ウエイト後、リターンします。次の場面に転換する間をとるものです
700〜790　CGの座標、ページデータ
800〜910　CGのデータ番号、ウエイト値、出力文字列です。最後の－１は、データの終了の意味です。データ数が可変の場合、このようにデータの最後を示すのも一つのテクニックです。

以上が今回のプログラムの説明です。次回では、また別な紙芝居の作り方を解説したいなーと考えています。

SAMPLE10.BAS

```
10 DEFINT A-Z
20 SCREEN 5:COLOR 15,0,0:CLS
30 DIM GX(8),GY(8),PG(8)
35 DIM DT(20),WT(20),C$(20)
40 OPEN "grp:" FOR OUTPUT AS #1
45 RESTORE 700
50 FOR I=0 TO 7
60 READ GX(I),GY(I),PG(I)
70 NEXT
80 RESTORE 800
90 I=0
100 READ DT(I),WT(I),C$(I)
110 IF DT(I)<>-1 THEN I=I+1:GOTO 100
120 SET PAGE 0,1
130 BLOAD"gdata1.ge5",S
140 COLOR=RESTORE
150 SET PAGE 0,2
160 BLOAD"gdata2.ge5",S
170 SET PAGE 0,0
200 'めいんふろぐらむ
210 D=0
220 GN=4:S$="":GOSUB 500
230 IF TIME<200 THEN 230 ELSE TIME=0
240 GN=DT(D):S$=C$(D):W=WT(D)
250 IF GN=-1 THEN 300
260 GOSUB 500
270 D=D+1
280 GOTO 240
300 CLS
310 PSET (8*11,8*13)
320 PRINT #1,"お し ま い"
390 A$=INPUT$(1)
400 END
500 'ひょうじサブルーチン
510 X=GX(GN):Y=GY(GN)
520 LINE (0,8*20)-(255,199),0,BF
530 COPY (X,Y)-(X+127,Y+105),PG(GN) TO (
64,50),0
540 PSET( 8*7, 8*20 ),0
550 FOR I=1 TO LEN( S$ )
560 PRINT #1,MID$(S$,I,1);
570 IF TIME<W THEN 570 ELSE TIME=0
580 NEXT
590 IF TIME<50 THEN 590 ELSE TIME=0
595 RETURN
700 'ざひょうデータ
710 '      x   y  page
720 DATA 000,000,1
730 DATA 128,000,1
740 DATA 000,106,1
750 DATA 128,106,1
760 DATA 000,000,2
770 DATA 128,000,2
780 DATA 000,106,2
790 DATA 128,106,2
800 'シナリオデータ
810 '     CG,wait,セリフ
820 DATA 0,10,あっ!ながればしだ!
830 DATA 3,15,どんな ねがいを かけたんだい.
840 DATA 1,15,ううん.まにあわなかったの!
850 DATA 5,10,あっ!まただ! えっと!なんにしようかな?!
860 DATA 3,15,もういいのかい?
870 DATA 1,15,ごめん. もういちどだけね!
880 DATA 6,10,えっと どうしようかなー. あれっ?
890 DATA 7,2,あっ! こっちにくる!!
900 DATA 2,2,うわぁぁぁぁぁぁ!!!
910 DATA -1,0,!
999 SAVE "sample10.bas",A
```

Mファンにいいたい放題！

★今月号はファンダムでターボRでしか動かないものや、パソ通天国でもDOS2じゃないと動かないものがあったので残念だった。ファンダムは1画面も好きだけど、D部門やツールを入れてほしいです。オールディーズはおもしろかったです。（富山市／中村秀紀・18歳）★ファンダムの『かみなりおとし』が良いです（最近疲れている）。あの短いプログラムであの楽しい動きは、驚きに値します。あれこそまさにファンダム（おもしろいけれど、商品にはなりえない）。いろいろと苦しいことが多いですが、これからもがんばってください。（東京都／秋本一郎・？歳）

Mファンにいいたい放題!

★8月号を読んで、本当にMSXがやばいことがわかった。すでにGTは入手困難になった(去年買っておいてよかった)。たとえMファンが入手困難になろうと値段が上がろうと必ず買いつづけるので、これからもがんばってMファンを作り続けてください。(埼玉県/荒木義博・16歳)

AFTER SCHOOL

紙芝居倶楽部~投稿作品より~

【中津さんと僕】

三重県/FIL-LY(16歳)
SCREEN5・4分の1画面サイズ

今回、またまた作品が減っちゃいました。シクシク。作品は引き続き募集しておりますので、どんどん送ってくださいね。

では、今回の採用作品です。三重県、ペンネームFIL-LYさんの作品「中津さんと僕」です。今月のサンプルプログラムと同様に、前々回の紙芝居をベースにした作品で、DATA文を使ったテクニックと、痛いところをついたセリフとで採用となりました。

◆ ◆ ◆

「はじめまして、初投稿です。

~「中津さんと僕」ストーリー~

説明するほどの話ではありませんが、とりあえず。MSX版サーク3の発売を願いながらも買おうとしない1ユーザーに中津さんの怒りの一発が飛ぶといった感じですね。

ここで1つ中津さんにお願い。この作品を採用して下さい。それがダメならMSX版サーク3を発売して下さい。こんなわがままな僕の願い、聞いてくれたらうれしいな」

と、いうことでMSXサーク3は、ごめんなさい!! としかいえませんが、PCエンジンのサーク3がNECより9月24日発売予定です。くわしくは、PCエンジンFAN(CM)などの雑誌をご覧ください。

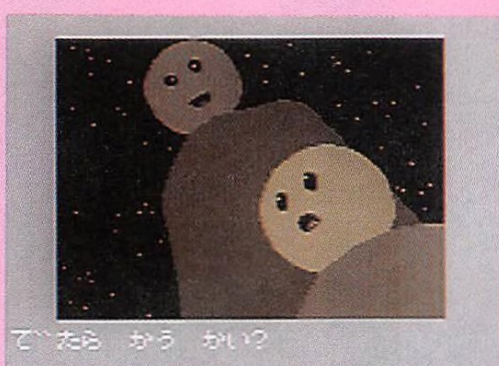
改造されているポイントは、セリフがDATA文に格納されている点でしょう。ちょとした改造ですが、プログラムを理解している事がうかがえます。

ただ、せっかくセリフをDATA文にしたのに、まだいくつかプログラムのとちゅうにもセリフが残っています。ぜひこれらのセリフも工夫してDATA文にまとめてみてください。

まず、行30でセリフ格納用の配列宣言の数値を変更。次に実際にセリフを配列に格納しているのが行80~行100のループなので、ここを少し変更します。そして行490のあとに新しくDATA文の行を追加するという要領です。

力のあるひとは、行220~行350の紙芝居表示のプログラムを、今月のサンプルを参考にまとめてみるのもいいでしょう。

内容も鋭い突っ込みに思わず笑ってしまいました。じゃ、他の投稿もどしどしお待ちしておりまーす。よろしくね。

⬆これが中津さんの痛いところをついたセリフ。採用の決め手となったようだ

⬆6-7月号の「星に願いを」とグラフィックは同じだが、セリフやタイミングは工夫してある

ナカツサン.BAS

```
10 DEFINT A-Z
20 SCREEN 5:CLS
30 DIM GX(4),GY(4),C$(3)
40 OPEN "grp:" FOR OUTPUT AS #1
50 FOR I=0 TO 3
60 READ GX(I),GY(I)
70 NEXT
80 FORN=0TO3
90 READC$(N)
100 NEXT
110 SET PAGE 0,1
120 BLOAD"a.ge5",S
130 COLOR=RESTORE
140 TIME=0
150 SET PAGE 0,0
160 'ﾒｲﾝﾌﾟﾛｸﾞﾗﾑ
170 CLS
180 PSET( 64 ,8*10 )
190 PRINT #1,"◆◆ なかつさん と ぼ゛く ◆◆"
200 IF TIME<300 THEN 200 ELSE TIME=0
210 CLS
220 FORN=0TO1
230 'no.1,3
240 I=1:C$=C$(N)
250 GOSUB 500
260 'no.2,4
270 I=3:C$=C$(N+2)
280 GOSUB 500
290 NEXTN
300 'no.5
310 I=1:C$="ううん。ともだ゛ち に かりる!"
320 GOSUB 500
330 'no.6
340 I=2:C$="ゆるさぁーん!"
350 GOSUB 500
360 CLS
370 PSET(108,100)
380 PRINT#1,"お わ り"
390 A$=INPUT$(1)
400 END
410 'さ゛ひょうデ゛ータ
420 DATA 0,0
430 DATA 128,0
440 DATA 0,106
450 DATA 128,106
460 DATA なかつさん!
470 DATA MSXで゛ サーク3 だ゛してよ!
480 DATA なんだ゛い?
490 DATA で゛たら かう かい?
500 'ひょうし゛サブ゛ルーチン
510 X=GX(I):Y=GY(I)
520 LINE (0,8*20)-(255,199),0,BF
530 COPY (X,Y)-(X+127,Y+105),1 TO (64,50
),0
540 PSET( 8*6,8*20 )
550 PRINT #1,C$
560 IF TIME<200 THEN 560 ELSE TIME=0
570 RETURN
```

中津さんとホームルーム

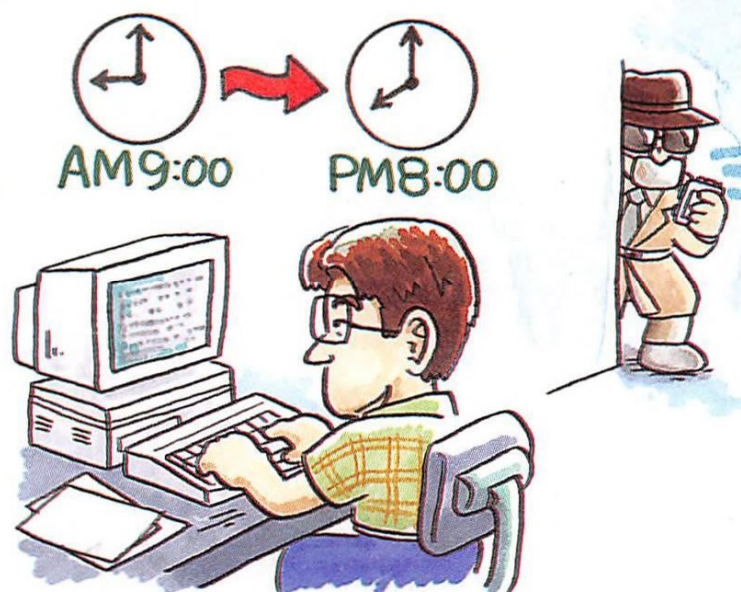
前回に引き続いてのマイクロキャビン情報コーナーだ。

おまたせした。わたしは、私立探偵のギャブ・モブラード。某機関の要請で、中津の行動を調べている者である。2か月振りとなってしまったが、最新のレポートをまとめたのでここに報告しようと思う。

さて、前回3Dポリゴンを応用した新しいSRPGを作っているという報告をしたが、その後さして目立った進展はしていない。連日仕事に励んでいるようだが、まだまだ形にはなっていないようだ。

わかっていることは、朝の9時ごろから夜の8時ごろまで仕事をしていること。世間のイメージよりは、意外と健全な日常を送っているらしいことだけである。

しかし、休日のはずの土曜日も仕事していたりと、ゲームを作ることはやはり、健全な日常生活を送るには無理があるようだ。

以上報告を終わる、次回こそは、その活動内容に深いメスを入れてみたいと思う(ザーッ)。

今月のPOINT／興味を引くタイトル画面を作ろう

タイトル画面は、作品の顔。その出来如何で、おもしろい作品もつまらなくなるし、その逆もありえます。そこでタイトル画面用にいくつかの技を紹介しましょう。もちろんタイトル画面以外にも応用は利きますよ。

→P85の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

タイトル画面に凝ってみよう

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の「サーク」でデビュー。以後「FRAY」、「幻影都市」などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

これがお手本 サークⅡタイトル

どもっ／　中津です。毎回いってますが、何でこんなに忙しいんだ〜っ／　ハァハァ。

最近徹夜モードでプログラムしていたので、ちょっとハイになってますが、よろしくおつきあいください。

紙芝居でもゲームでも、タイトル画面は、とても大事です。本でいえば表紙にあたります。それが面白ければ、作品自体に興味がわくでしょう。また同じレベルの作品なら、タイトルの出来のいいほうが、印象に残るでしょう。

紙芝居のタイトルということなら、もちろん一枚の絵でもいいのですが、よっぽどの絵でないと人の興味を引くことは難しいでしょう。そこで、せっかくの動画のテクニックを使わない手はありません。

さて、今回は忙しかったので、あまり時間をかけていられませんでした。ごめんなさい。で、時間がないときの必殺技、身内のネタばらしをやることにしました。

ネタは『サークⅡ』のタイトルもどきです。

と、今回はCG絵師末永先生に頼んでCGを描いてもらいました。いかがでしょうか？　明日までねぇ〜。サイズはこんだけぇ〜とムチャをいってしまいました。反省してます。

ではさっそくプログラムの説明に入りましょう。

サンプル1 サークⅡもどき

プログラムは前半と後半に分かれています。前半は行10〜60、後半は行110〜150です。

それでは行ごとに詳しく見ていきましょう。

⚜　⚜

10　変数を整数型に宣言
20　SCREEN5に初期化
25　文字色、地の色などを設定して画面クリア
30　裏の画面（ページ1）に対してグラフィックをいじるためのおまじないです
40〜50　末永絵師の画像データを読み込んでいます
60　表の画面（ページ0）に対してグラフィックをいじるためのおまじないです
110　Iは元データのY座標で、上から下へ少しずつ転送します
120　Jは転送先のY座標で下から上へと上がっていきます。Iをループの終値に使っているのは、最終的に上がりきったポイントで元の絵を復元させるためで、だんだんとIに合わせて下げていくのです
130　Iの示す座標から縦に3ドットのブロックをJの示す座標にCOPY命令で転送します
140〜150　画像をすべて転送しおわるまで繰り返します

⚜　⚜

とこれだけのことで、サークⅡのタイトルもどき処理ができてしまいます。前半は単なる準備なので、実際に絵を動かしている部分は行110〜150のたったの5行しかありません。プログラムの短さに驚かれた人もいるのではないでしょうか。

ガゼルの塔のタイトルもどきも載せようかと思ったのですが、BASICでは遅すぎて見本になりませんでした。これはまたの機会とさせていただきます。

SAMP12A. BAS

```
10 DEFINT A-Z
20 SCREEN 5
25 COLOR 15,1,0:CLS
30 SET PAGE 0,1
40 BLOAD "mfan12.ge5",S
50 COLOR=RESTORE
60 SET PAGE 0,0
100 '12ｶﾞﾂｺﾞｳｻﾝﾌﾟﾙ1 ｻｰｸ2ﾀｲﾄﾙﾌｳ
110 FOR I=8 TO 96  STEP 2
120  FOR J=224 TO I+48 STEP -2
130   COPY (16,I)-(176,I+2),1 TO (48,J)
140  NEXT
150 NEXT
999 'SAVE "samp12a.bas",A
```

SAMP12A. BAS実行

サンプル2 左右スライド

さて、続いてはタイトル風の動画処理その2です。左右からスライドしてきた絵が合わさって1枚の絵になるプログラムです。

5〜20　初期化です

30〜50　グラフィックをページ1にロードします

60　READ文で読み込むデータのある位置をRESTORE命令で行500から始まるDATA文に宣言をします

110　LLはあつかうグラフィックの縦サイズでL1はグラフィック命令で使用する座標用に−1しておきます

120〜130　DATA文から表示位置をどれくらいずらすかのデータをILに読み込みます。データが−1ならプログラムを終了します

140　Iは0から92の値でLL単位で変化します。転送元のY座標に使用します

150　ICはスイッチで、−1でON、0でOFFです。ONならIPに32−ILの値、OFFならIPに32+ILの値を入れます。IPは転送先のX座標として使用します

160　IPが0未満なら特殊な処理のため行170へ行きます。0以上なら行200へ行きます

170〜180　COPY命令では、負の座標へ転送すると処理がおかしくなるので、画面に表示される部分だけを転送するようにします

200　Iの示す座標からIPとIの示す座標に転送します。右へはみ出たぶんはVDPが処理してくれるので特殊なことをする必要はありません

210〜220　スイッチのICを反転して繰り返します

230　次のデータ処理のため行120へ戻ります

510　変化する座標のデータです（下のカコミ参照）

520　終了のデータです

SAMP12B. BAS実行

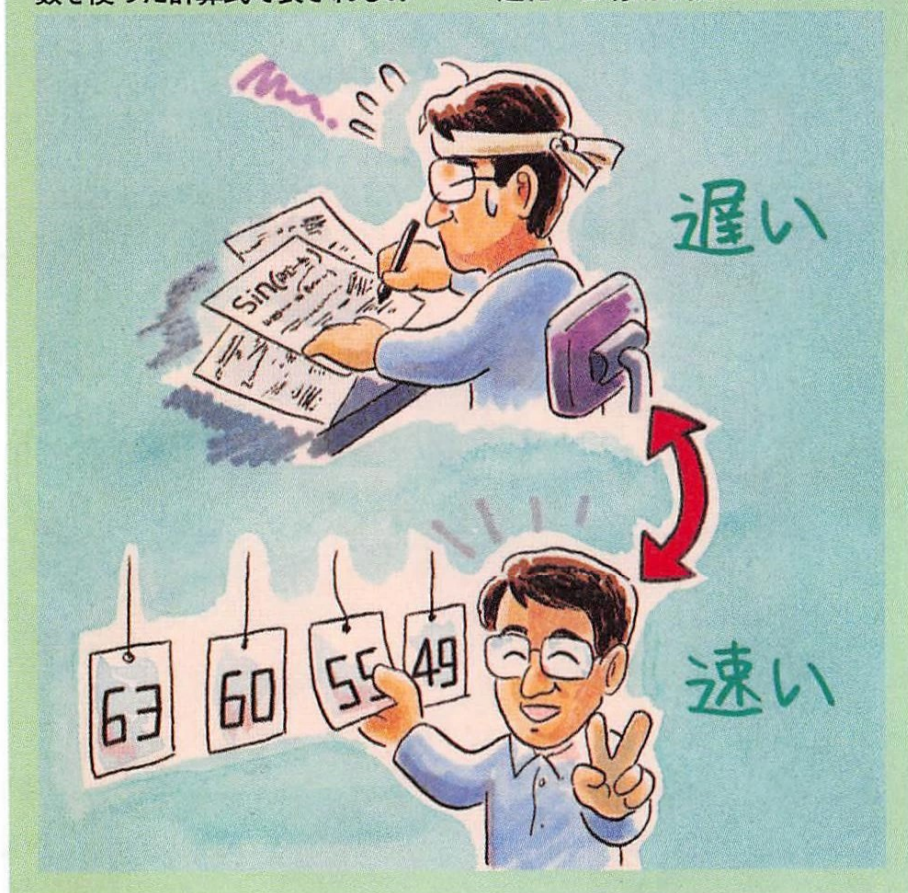
SAMP12B. BAS

```
5 DEFINT I-M
10 SCREEN 5
20 COLOR 15,1,0:CLS
30 SET PAGE 0,1
40 BLOAD"mfan12.ge5",S
50 COLOR=RESTORE
60 RESTORE 500
100 '12か゛つこ゛うｻﾝﾌﾟﾙ2 ｽﾗｲﾄﾞ ﾀｲﾌﾟ
110 LL=8:L1=LL-1
120 READ IL
130 IF IL=-1 THEN END
140 FOR I=0 TO 92  STEP LL
150 IF IC THEN IP=-IL+32 ELSE IP=IL+32
160 IF IP<0 THEN 170 ELSE 200
170 COPY (-IP,I)-(178+IP,I+L1),1 TO (0,I
+64),0
180 GOTO 210
200 COPY (0,I)-(178,I+L1),1 TO (IP,I+64)
,0
210 IC=NOT IC
220 NEXT
230 GOTO 120
500 'sin ﾃﾞｰﾀ
510 DATA 63,60,55,49,41,32,21,11,0,0,0
520 DATA -1
999 'SAVE "samp12B.bas",A
```

テーブルを使おう

左下のリストの行510を見てください。このDATA文はIL（表示位置をどれだけずらすか）に次々に読み込まれていきます。ここを変更すると左右の動きが変わります。

一般的に自然な動きは二次関数や三角関数などの数式で表されます。例えば落下は二次関数です。今回のILは、SIN関数を使った計算式で表されるのですが、これをそのままプログラムにすると、実行時にとても時間がかかります。

そこで、計算結果をあらかじめデータとして持っていて、それを順番に読み込んで使う、という技を使います。このようなデータを「テーブル」といって、その数値を使うことを「テーブルを参照する」といいます。高速化には有効な技ですよ。

Mファンにいいたい放題！

★はっきりいって、もっと早く終わると思っていた。……あっ、これは悪い意味ではなく、世間の常識からいくと、人気のない（古い）ものは市場から消える→売れない→専門誌は営利目的のため（ボランティアではない）運営しているので、休刊もしくは廃刊となる。しかし、Mマガが撤退してもMファンはがんばってくれた。私がMファンを知ったころはフリーターの青二歳だったが、今は30歳である。あの国民標準機であるPC-98の専門誌でさえ休刊があいついでいるのに、お偉いさんからも「休刊しろ〜」といわれ続けたろうに（?）、今までがんばってくれて本当にありがとうございます。（東京都／ねこ太郎・30歳）

Mファンにいいたい放題！

★『ディスクステーション#32』でFINALだと知ったとき、頭が真っ白になったことがあったが、MSX・FAN10月号の最後のページをみてしまったときも、本を持ったままそういう状態だった。ところで、TBS系列の番組だったと思うが、昔パソコンを使ったゲーム番組があった。キーボードの1〜3を使って赤・青・黄色のインベーダーを消す、崩れていく土台から逃げながらロケットに乗る、最後に空中ブランコ、という組み合わせだったが、その機種がMSXだった気がする。いずれにせよ、今は事情が違うようだ。そんな時代が戻ってくることを願う。（新潟県／大鷹孝・18歳）

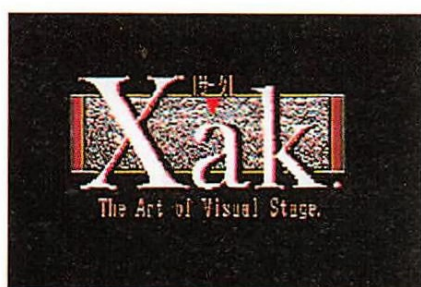
実際のゲームのタイトル画面は

さらに手前ミソですが、どこかで見たようなゲームのタイトル画面を並べてみました。どれがどのゲームか、わかりますか？　これらのタイトルは、今まで講義で紹介してきた技の応用で、たいていできてしまいます。例えば、サークは重ね合わせスクロールの応用です。ただ、ガゼルの塔の文字が波のように揺れる技（ラスタースクロールといいます）は少し難しいので今回は割愛します。

他にもパレットを変化させたり（フラッシュやフェードイン・アウトなどに使います）、画像の一部だけ表示してその表示位置を少しずつ変えていったり（パンといいます）と、そう難しくない技でも工夫次第で見応えのあるものが作れます。

投稿する紙芝居でも、ちょっとした動きだけでいいですから、少し動きを入れてみると印象が違ってくるでしょう。採用される確率もさらにアップすることうけあいですよ。

ただし、タイトル画面ばかりに気を取られて、作品自体がおざなりにならないようにしてくださいね。

さて、ついにMファンも今号を合わせてあと5冊となってしまいました。MSXと長いつきあいの私も非常に残念です。あと4回精一杯頑張りますので、ご協力をお願いいたします（編集部注：投稿の催促のようですよ）。

タイトル文字の背景の矩形（くけい）の中で動いている模様がアクセントに

タイトル文字が波のようにうねる

揺れが収まりガゼルの塔の文字が

タイトルはちょっとおとなしめ

夜景に浮かび上がってくる文字

文字の中に炎が揺れている

中津さんとホームルーム

わたしは、私立探偵のギャブ・モブラード。中津の行動を調べている者である。

中津が作っているRPGだが、なかなかガードが堅くて、詳しいことがわからなかった。そのかわりに、末永という者の情報が入手できた。

末永はこのところ艦隊モノのシミュレーションゲームのCGを描いていたようだが、このゲームの名前が判明した。『紺碧の艦隊』という、原作は荒巻義雄の仮想戦記のヒット作品（徳間書店刊）だ。機種は残念ながらPC−98のようで、中津たちもMSXでなくて残念そうだ。

以上で今回の報告を終わる。

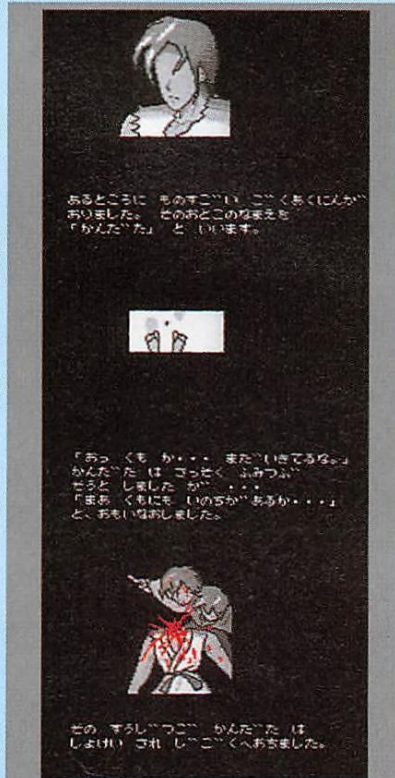
AFTER SCHOOL

紙芝居倶楽部〜投稿作品より〜

［蜘蛛の糸］

福岡県／村上周太（20歳）
SCREEN5・9分の1画面サイズ

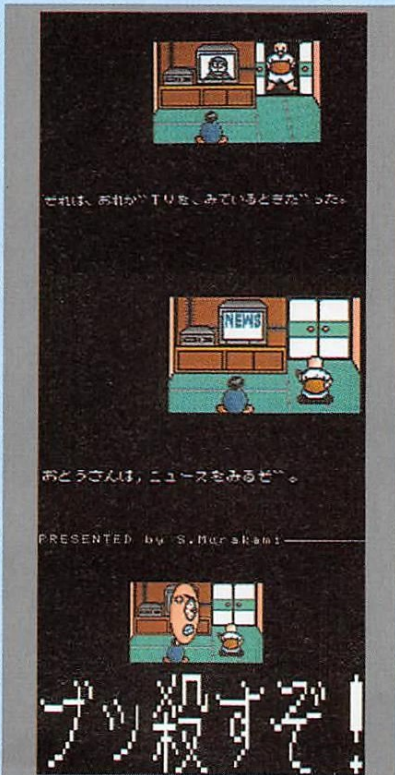
この作品は、絵が切り替わって話が進む、紙芝居の見本のような作品です。おしむらくも残念な点は、ギャグかと思わせる始まりであったのに、強烈なオチがなかったというところでしょう。マジメにやるならば、有名な話だけに最後の部分にもっと余韻が欲しかったです。

〈ストーリー〉あるところに「かんだた」という極悪人がいて、処刑され地獄に落ちた。彼は生前に蜘蛛を助けるという善行をしたおかげで、その蜘蛛の糸が極楽から垂れてきたが……。芥川龍之介の有名なお話。

［親父］

福岡県／村上周太（20歳）
SCREEN5・9分の1画面サイズ

上の作品と違って、こちらは別方向で非常にアクティブな動画作品。端的でわかりやすく、ギャグオチも決まっています。

プログラム的にもシンプルにできているので、見本としても最適と思いますが、COPY命令を並べてある部分の座標等をDATA文で処理できれば、もっとすっきりします。次回はそちらにも挑戦してください。

〈ストーリー〉僕がテレビアニメを見ていたところに、親父がズカズカ入ってきた。親父はリモコンを取り、「ニュースを見るぞ」といって……。

紙芝居&動画教室

今月のPOINT／マスクを使ったフェードイン・アウト

RPGで枠やステータス情報を残しつつ、プレイ画面だけをフェードイン、フェードアウトするときにはどうしたらいいだろうか。というときに、今回のテクニックが使える。マスク処理は非常に応用の利くテクニックである。

→P84の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

マスクを使おう

あけましておめでとう／　といっても、この原稿を書いているのは11月の中旬なので、実感がわきません。年末進行というやつで、こんな早い時期に書いてもみんなが読むのは、年が明けた今です。

さて、今回のサンプルは、CGをフェードインするのにパレットを使わずに、ドットの操作だけでやってみようというものです。

CGは末永先生にお願いした入魂の一作。楽しんでもらえると思います。

まず、フェードイン、フェードアウトの説明ですが、くわしいことは右下のカコミでするとして、要はだんだんと絵が現れたり消えたりすることと、ここでは認識してください。

フェードインをする方法は、いろいろと考えられますが、今回はマスクという手法を使ってみましょう。

絵をドット単位で隠してしまう､網のようなものを作って､これを絵にかぶせて、一定の割合で元の絵が見えなくなるようにします。こうすると人間の目には、絵がぼやけたり、暗くなったように見えてしまうんですね。

この網を「マスク」、網を絵にかぶせることを「マスクする」とか「マスクをかける」といいます。

網の目が大きくなっていくマスクをいくつか用意しておいて、元の絵がだんだんと見えるようにマスクをかぶせていけば、フェードインのできあがりです。

マスクを作るためには、COPY文のAND機能を使います。ANDとは論理演算で（1と0だけの演算と思っていいです)、1AND1なら1になりますが、それ以外はすべて0になるというものです。ここでわからなくても構いませんが、ANDはマスクで使う機能と覚えてください。

マスクデータは、カラーコード0と15で作ってあります。残す部分はカラーコード15で、消す部分はカラーコード0を使います。2進数で15は1111、0は0000なので、ANDを使うと、AND15なら残りますが、AND0なら消えるというわけです。15以外を使うと、赤い色だけ残すなどという技もできます。いろいろ考えてみてください。

マスクの処理について

マスク処理をかけるとマスクの形に絵が残る。CGのマスクの場合、マスクの形やカラーコードを変えるだけで様々な効果も出せる。

フェードイン・アウト

フェード（fade）というのは、消えるという意味の英語。そしてフェードイン、フェードアウトとは、それぞれ溶明、溶暗という映画用語で、光のなかや暗闇のなかに、姿が溶けるように見えなくなることです。しかし、本来は間違いなのかもしれませんが、フェードインを次第に姿を現すという意味で使うこともあるようです。

←今号の付録ディスク、ルーシャオの冒険の一場面

←パレット操作でフェードアウトしていく

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の『サーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

これを応用したのが、サンプルAとサンプルBのプログラムです。

サンプルA（フェードアウト型）は、元の絵がぼやけていき、新しい絵と入れ代わります。

サンプルB（ワイパー型）は、元の絵が右から消えていくとともに、右から新しい絵が出現してくるものです。

元の絵がなければ単なるフェードイン。新しい絵がなければ普通のフェードアウトというところでしょうか。

付録ディスクを解凍すると、それぞれ“SAMP2A．BAS”と“SAMP2B．BAS”というプログラムが出てきます。また、2つのプログラムで共通に“MFAN02．GE5”というBSAVE形式のグラフィックを使っています。

このグラフィックのなかのマスクCGの部分を描き変えると、また違った効果のフェードイン、フェードアウトになるので、CGツールを持っている人は試してみるといいでしょう。

‡　‡　‡

ところでだ（……と口調も変わる）。

今作っているゲームがちょうどこの本の発売日あたりにマスターアップする予定だ。無事正月をすごせるかどうかは、今んとこレッドゾーン（笑……えない）だが、おもしろいゲームを作るためにがんばるつもりだ。期待してくれ。

マスターアップとは、完成したプログラムを、CD-ROMにしてくれるメーカーに納品することだ。マスターアップ後、1～2か月して発売となる。

開発中のゲームは、次世代機セガサターンで遊べるソフトだ。くわしい内容はサターンFANを見てくれるとうれしいぞ。

では投稿のほうもよろしく頼む。かんたんな紙芝居から、ひねりを利かせたアニメーションまで、待っているよ。

Mファンにいいたい放題！

★以前、アンケートはがきにこう書いたことがあります。「もしMファンが終わってしまったら、僕はMSXを手放すつもりです。自分ひとりでMSXを使いこなしていく自信がないからです……」。そのときは、本当にそう思っていました。もっとも、当時はMファンが本当になくなるなんて考えてもみなかったので、こんなことが書けたのでしょう。でも、もちろん今はMSXを手放すなんて考えていません。現在のMSXに必要なものは何でしょうか。それは新ハードでもゲームソフトでもありません。もっとも必要なものは、MSXユーザーの存在、そして、ユーザーの熱意です。これなくしてMSXの将来を憂慮するなど本末転倒です。もしMファンが終わる↗

フェードアウト型

ピクシーの絵がだんだんとぼやけていくと同時に、フレイの絵がだんだんと現れてくるサンプルです。

プログラム中では、ピクシーにはだんだんと消えていくマスクをかけて、フレイにはだんだんと残る部分が大きいマスクをかけて、それを合成して表示画面に転送します。

プログラム中で出てくる変数Bは、処理する画面のページ番号です。

‡　‡　‡

10～40　初期化です

50～80　グラフィックをページ1にロードしてパレット設定。表示画面設定

100　フェードインループ開始。マスク用のCGのX座標が0から27まで3ドットずつ変化します

110　ページ1のフレイとピクシーを、ページ2にある処理のバッファに毎回転送します

120　CGにマスク処理をするためのループです。X座標が0～60まで3ドットずつ変化します

130　マスクのCGを使ってバッファ上のフレイにマスクをかけます。Iが進んでいくにつれて、だんだんと絵が残るマスクが使われていきます。COPY文の最後に書かれているANDがその秘密です

140　マスクのCGを逆から使ってバッファ上のピクシーにマスクをかけます。Iが進むにつれて、だんだんと絵が消えるマスクが使われていきます

150　フェードインループ閉じ

160　バッファ上のピクシーをフレイの上に重ねます

170　ページ2のバッファ上のCGを、ページ0の表示画面に転送します

190　ウエイトです

210　行90に戻って再びデモを始めます

SAMP2A. BAS

```
10 DEFINT A-Z
20 B=2
30 SCREEN 5
40 COLOR 15,1,1:CLS
50 SET PAGE 0,1
60 BLOAD "mfan02.ge5",S
70 COLOR=RESTORE
80 SET PAGE 0,0
90 '2ｶﾞつこﾞうｻﾝﾌﾟﾙ1 ﾌｧｰﾄﾞｲﾝ
100 FOR I=0  TO 27  STEP 3
110  COPY (0,0)-(127,63),1 TO (0,0),B
120  FOR X=0 TO 60  STEP 3
130   COPY (I,64)-(I+2,127),1 TO (X,0),B
,AND
140   COPY (27-I,64)-(27-I+2,127),1 TO (
X+64,0),B,AND
150  NEXT
160  COPY (64,0)-(127,63),B TO (0,0),B,T
PSET
170  COPY (0,0)-(62,63),B TO (64,64),0
180 NEXT
190 FOR I=0 TO 2000:NEXT
200 CLS
210 GOTO 90
```

VRAMの裏では

画面に表示されないVRAMは、実行されるアニメの素材となるグラフィックが置かれていたり、マスク処理を行うための一時的な作業領域になったりと、頻繁に使われています。

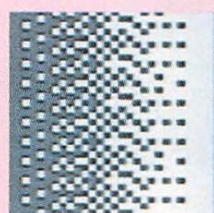
ページ2は、プログラムによって使われ方が決まっていますが、ページ1は素材となるグラフィックが読み込まれるだけなので、CGツールでいろいろいじってみてください。右の写真のように、左上からフレイとピクシーのCGが64×64ドット。その下に、マスク用のCGが3×64ドットで10種類描かれています。

ページ1

↑ファイルから読み込まれる。ここから各CGが処理のバッファに転送されていく

↓マスクCGの拡大図。右にいくにしたがって、残る部分が多くなっていく

ページ2

↑上の段が処理のバッファ。下の段がワイパー型で使うバッファ2である

ワイパー型

ピクシーの絵が右からワイパーによって消えてくと同時に、その跡地に右からだんだんとフレイの絵が現れてくるサンプルです。ワイパーの跡地の部分にマスク処理をしています。

行110のループがー30までなのは、ワイパーの跡地には幅があるため、それが通り過ぎてフレイの全身が現れるまで時間がかかるからです。

変数Bはやはり処理する画面のページ番号です。

‡　‡　‡

10〜40　初期化です
50〜80　グラフィックをSCREEN5のページ1にロードします。グラフィックを処理するページをBにセットします
90　バッファをクリアします
95　ピクシーをバッファの2に転送します
110　フェードインループ開始。X座標が右から左へ3ドットずつ変化するループです
120〜130　ワイパーがピクシーを全部消したあと、X座標がマイナスにならないようにガードします
140　処理するためのCGを、Xのしめす部分から転送します。マスク処理で壊れるため毎回転送します。だんだんと左へシフトしていきます
150　CGにANDのマスクをかけます。マスクのサイズは30×64ドットです。Iのしめす座標に合わせてだんだん左へ移動します
160　バッファ2に置いたCGをだんだん左へ消していきます
170　バッファ2の上にANDでマスク処理したCGをTPSETで重ね合わせます
180　合成したCGを表示ページ0に転送します
190　フェードインループ閉じ
200　ウエイトです
210　行90に戻って再びデモを始めます

SAMP2B. BAS

```
10 DEFINT A-Z
20 B=2
30 SCREEN 5
40 COLOR 15,1,1:CLS
50 SET PAGE 0,1
60 BLOAD "mfan02.ge5",S
70 COLOR=RESTORE
80 SET PAGE 0,B
90 LINE (32,0)-(94,63),0,BF,PSET
95 COPY (64,0)-(127,63),1 TO (0,64),B
100 '2か゛つこ゛うサンフ゜ル2 ファート゛イン2
110 FOR I=64 TO -30 STEP -3
120  X=I
130  IF X<0 THEN X=0
140  COPY (X,0)-(X+29,63),1 TO (32+X,0),
B
150  COPY (0,64)-(29,127),1 TO (32+I,0),
B,AND
160  LINE (X,64)-(X+2,127),0,BF,PSET
170  COPY (32,0)-(94,63),B TO (0,64),B,T
PSET
180  COPY (0,64)-(63,127),B TO (80,64),0
190 NEXT
200 FOR I=0 TO 1000:NEXT
210 GOTO 90
```

AFTER SCHOOL
紙芝居倶楽部 [ちりがみアニメ]

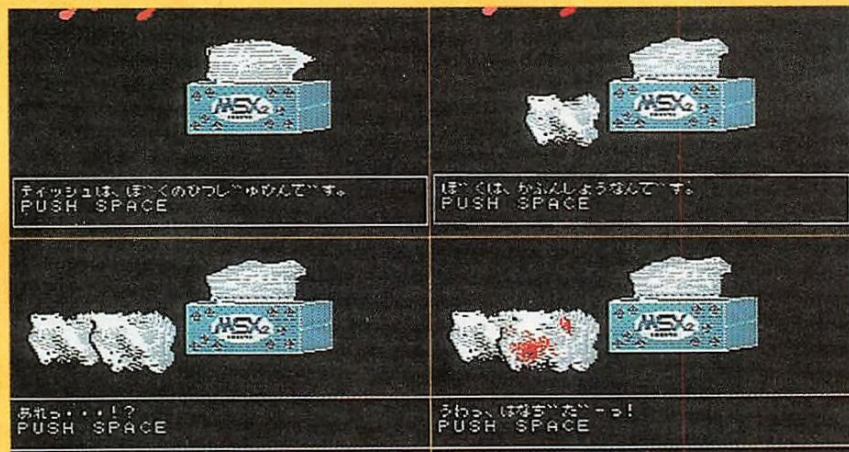
〜投稿作品より〜

白神裕久/愛知県・18歳
SCREEN5・4分の1画面サイズ

今回の作品は『ちりがみ』です。

押しの強い強引な展開が気に入ってしまいました。少ないCGをうまく合成することで、多量のちりがみの山を作ってしまったところも、紙芝居のプログラムとしてポイントが高いところでしょう。プログラムも、ループやDATA文をうまく使っていてまとまっています。

作者は、ちりがみを回転させてパターンを増やそうと試みたそうですが、速度の点から断念したようです。しかし、これはこれでうまくいっています。

次はぜひ、もっと強引な作品を作ってください。

花粉症ではなが止まらないらしく、大量のちりがみが散らかっていく

中津さんとホームルーム

トゥルルルル……ガチャ（電話の音らしい）。

私の名は、ギャブ＝モブラート。私立探偵だ。某機関の要請で、中津の行動を調べているものである。

さっそく、新しい情報が入ったので報告したいと思う。

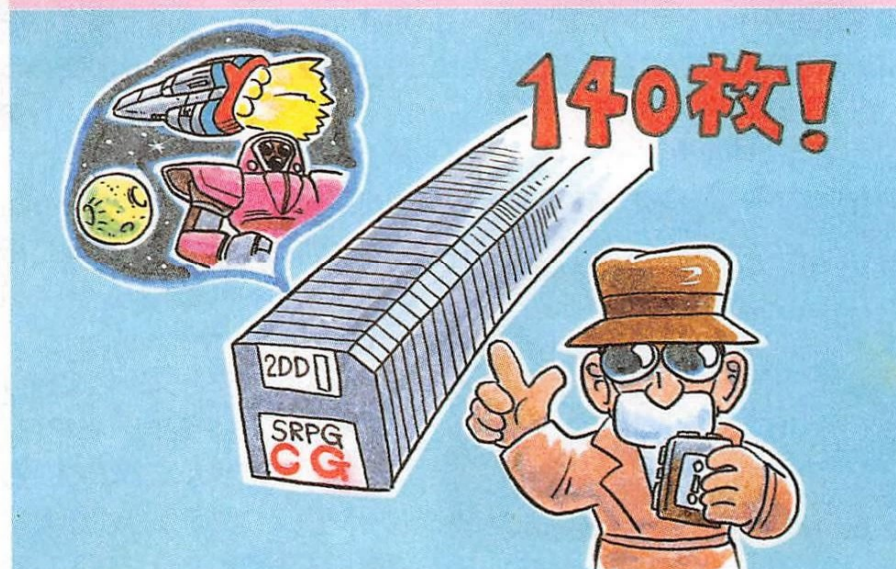
ウワサされていたSRPGであるが、なんとCGのデータだけで100MBytes以上もあることが判明した。

2DDのフロッピーディスクで140枚分ほどらしい。世の流れとは恐ろしいものである。

ただし、セガサターン用ということであり、ゲームの供給はCD-ROMになるので、容量的にはなんとかなるようだ。ちなみにCD-ROMの容量は540MBytesだ。

ところで、サターンのゲームとは『リグロードサーガ』という情報が入った。では、また。

Mファンにいいたい放題！

↘からという理由でＭＳＸを手放そうという人がいたら、ちょっと待ってください。前にいったとおり、ＭＳＸにおいて重要なのは、けしてＭＳＸ・ＦＡＮではなく（ごめんなさい）、ユーザーの力です。情報源が欲しければパソコン通信を始めればいいのですし、これから高校や大学に進学する人などは、そこでユーザー仲間を見つけるのもいいでしょう。ＭＳＸが終わっても、ＭＳＸがすたれるわけではありません。ＭＳＸもきっと、ユーザーのもとで使われ続けていきたい、と思っているのでは……？（福井県／坂下佳嗣・17歳）

紙芝居&動画教室

今月のPOINT/パレット操作に必要な命令

SCREEN5や7では色を表現するのにパレットを使っている。これを上手に変化させるだけで思わぬ効果が得られる。今回は現在のパレットから好きなパレットへだんだんと変化させていくサンプルで、その使い方を紹介する。

→P85の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

少しずつ色を変化させる

やー、どもどもども。お久しぶりです。2か月のごぶさたでした。

ちょうどこのMファンの仕事をしているときに、風邪を引いてしまい編集部にはご迷惑をおかけしました。ペコッ。

この本が出るころには、インフルエンザもおさまっているといいな。

さて、今回はカラーのCGを白黒に変化させるテクニックを紹介します。ちょっとややこしいことをしますが、完全に理解しなくてもテクニックとしては使えますよ。

サンプルを実行すると、おなじみフレイとピクシーがカラーで表示されます。そしてだんだんと彩度(あざやかさ)が低くなっていき、最後には白黒になってしまいます。

このプログラムのポイントは、VRAM中にたくわえられているカラーパレットのデータを取り出してきて、そのデータを元に白黒のパレットデータを作り出していることにあります。

新しい命令として、VPEEKとCOLORが出てきます。

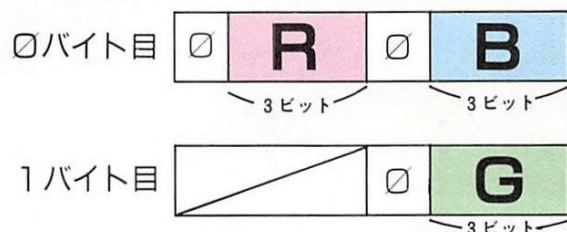
命令の意味は右上のカコミの通りで、VPEEKはVRAMのパレットを読み出すのに使われています。VRAMのアドレスはスクリーンモードによって違います。くわしくはMSX2テクニカル・ハンドブックなどで確かめてください。

さて、今回の紙芝居倶楽部では、4作品を採用しました。何せ応募が多かったもので、思わず嬉しくなってしまっての多量採用です。

しかし、ついに最後の投稿の募集になってしまいました。8月号の募集の締め切りがだいたい5月始めになりますので、今までの講義で覚えた知識を総動員して、中津を驚かせる紙芝居を送ってください。これが最後のチャンスです。くれぐれも、締め切りに遅れないようにね。

VRAM中のカラーパレット

SCREEN5の場合、VRAMのアドレス&H7680からの32バイトにカラーパレットの値が保存されています。

図のように1色2バイトで16組あり、計32バイトというわけです。SCREEN5以外はVRAMのアドレスが変わります。

[VPEEK]

VRAMの任意のアドレスから1バイトのデータを取り出す。

例:A=VPEEK(&H7680)

VRAMの16進数で7680のアドレスにある1バイトのデータを取り出して変数Aに代入する。

[COLOR=(N, R, G, B)]

カラーパレットを任意のカラーパレット番号(N)ごとに、512色のなかからRGBの値を指定して変更することができる。

例:COLOR=(1, 5, 5, 5)

カラーパレット番号1の色をR(赤)=5、G(緑)=5、B(青)=5の灰色にする。

SAMP4.BAS

```
100 DEFINT A-Z
110 DIM R(16),G(16),B(16),BW(16)
120 SCREEN 5
130 COLOR 15,1,1:CLS
140 SET PAGE 0,1
150 BLOAD "mfan02.ge5",S
160 COLOR=RESTORE
170 COPY (0,0)-(127,63),1 TO (64,64),0
180 FOR I=0 TO 15
190  V=&H7680+(I*2)
200  B(I) = VPEEK(V) AND &HF
210  R(I) = (VPEEK(V) / 16 ) AND &HF
220  G(I) = VPEEK(V+1) AND &HF
230 NEXT
240 FOR I=0 TO 15
250  W = B(I) + R(I) + G(I)
260  BW(I) = W / 3
270 NEXT
280 FOR J=0 TO 15
290  FOR I=0 TO 15
300   IF BW(I) > R(I) THEN R(I)=R(I)+1
310   IF BW(I) < R(I) THEN R(I)=R(I)-1
320   IF BW(I) > G(I) THEN G(I)=G(I)+1
330   IF BW(I) < G(I) THEN G(I)=G(I)-1
340   IF BW(I) > B(I) THEN B(I)=B(I)+1
350   IF BW(I) < B(I) THEN B(I)=B(I)-1
360   COLOR=(I,R(I),G(I),B(I))
370  NEXT
380 NEXT
390 GOTO 390
```

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の『サーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

それでは順にプログラムを見ていきましょう。

まず変数の初期設定をしています(行100)。R、G、Bの配列はパレットをたくわえるもので、BWは白黒のパレットを入れるためのものです(行110)。

つぎにグラフィックを初期化します。おなじみのSCREEN5で、BLOADでグラフィックをロードします。実はグラフィックは前回とおなじものを使用していますので、表示位置をずらしたりして(行170)、ちょっと特殊なプログラムになっています。他のグラフィックをロードする場合、行140と行170を削除していただければOKです(行120〜170)。

さらにVPEEKでVRAMのアドレス&H7680〜&H769Fにあるパレットデータを取り出します(行180〜230)。データはR、G、Bの配列に分離して代入します。

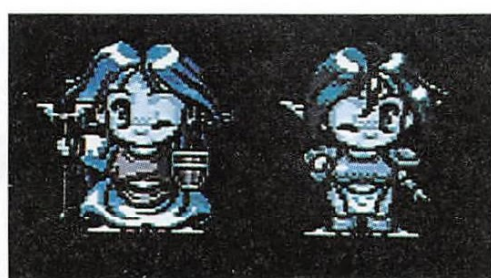
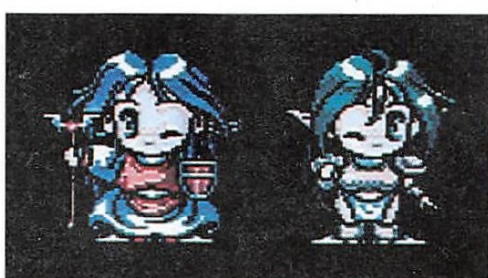
準備の最終段階としてR、G、Bのデータを元に変化後の白黒のデータを作り出します(行240〜270)。R+G+Bを3で割って平均化したものが白黒データとしてBWに代入されます。

ここからが本番です。パレットを徐々に変化させて白黒にします(行280〜380)。変数Jのループは変化させせるためのもの。15回ループしていますが、本当は7回で十分です。変数Iのループはパレットの16色分のループ。

6つ並んでいるIF文は、R、G、Bを各々変化させて、白黒に1ずつ近づける処理をしています(行300〜350)。COLOR文はパレットをセットする命令。

今月のサンプル

Mファンにいいたい放題！

★僕はMSXI（カシオのPV-16を改造してRAM32K）しか持っていない上に、クイックディスクで、付録ディスクも使えません（ヒサン）。たくさんのおもしろそうなゲームを目の前に、1画面プログラムだけを入力しています。昔のMファンにはリストがあったそうですが、僕はそのほうがよかったかな。MSXIでも1画面なら投稿できるかもしれません。しかし、せっかく12月号から買い始めた（書店で偶然見つけた）のに、もうすぐ休刊なんて、残念でなりません。（京都府／川北将・15歳）★「あと○冊で休刊」とMファンがちゃんといってくれたことは（変ないいかたかもしれませんが）本当によかったと思っています。おかげで、自分でも気持ちの↗

AFTER SCHOOL 紙芝居

【A. Senna Memorial】

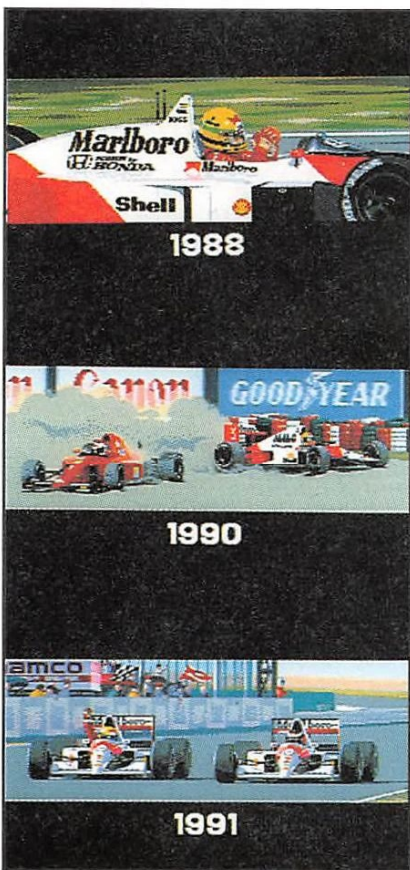
Fly☆Duck／青森県・29歳　SCREEN8・全画面サイズ

前回のCGコンテストで見たことのあるCGも混じっていますが、それらをうまくビジュアル的にまとめた、セナへの愛が見える作品です。

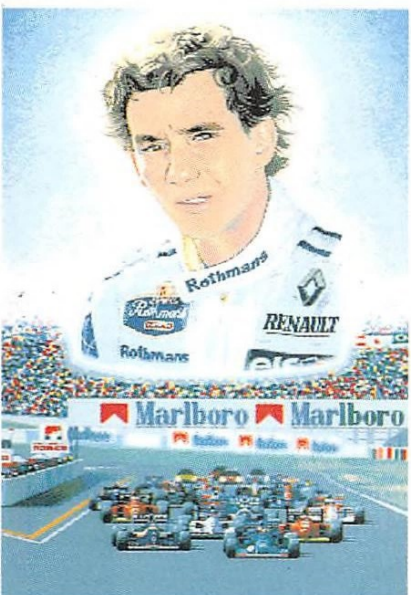
セナが過去3回ワールドチャンピオンを取ったシーンが手描きとは思えない実写取り込みのような質のCGで再現されます。

私もあの日は仕事で徹夜をしていたのですが、たまたま息抜きにアレを見て驚かされた記憶があります。

BGV風にまとまっていて、良い雰囲気です。

プログラムでターボRや機種を判別して、PCMを使って効果音を出したり、RAMディスクを使って読み込む回数を減らしたりと、いろいろな工夫もあります。

過去にセナが優勝を決めた場面。ターボRでは最後にPCMでセナの走行音が響く

前号のCGをリンク！

この作品は、95年2月号のCGコンテスト「セナのいない週末」を合わせて完全になります。前回のCGコンテストから、SENNA0.SR8、SENNA1.SR8、SCRSYS.BINの3ファイルを今回の紙芝居のディスクに移して実行してください。

【これでいいのだ。】

星野ぴかそ／神奈川県・年齢不詳　SCREEN5・4分の1画面サイズ

一発ギャグ押しの作品です。どこが、これでいいのだ。か、何が何だかわかりませんが、思わず納得してしまう作品でもあります。編集部の福田氏の強いプッシュもあっての採用です。

紙を広げる直前に、ちょっと間をとっているのがうまいぞ。

プログラムもシンプルで、たいへんよろしい。花マルをあげちゃいます。

実はこの作品、前回の2月号ぶんに投稿されたもので、一発ギャグを載せるコーナーじゃないからという理由で、一度は落選したのですが、前担当の福田氏たっての希望で今回の採用と相成りました。それでは、編集部の福田氏の推薦の声もどうぞ。

福田「ごっつ〜ええわ。間といい構図といい表情といい、最高。無罪放免といえば、どんO ぃ氏の同人誌じゃけんど、これは、無……ンググ」(ないしょだってば)

これでいいのだ。

バーン！　という擬音が聞こえてきそうな地方裁判所のアオリ

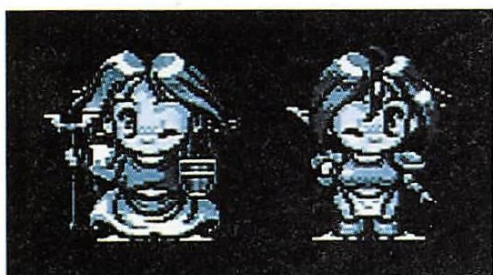
その裁判所の玄関。そろそろ裁判長から判決が言い渡されるころだ

第一報を手に出口から駆け出してきた。さあ、吉報か凶報か……

歓喜あふれる笑顔とともに、手にしたかみには「無…」の文字が見えた

ないしょ

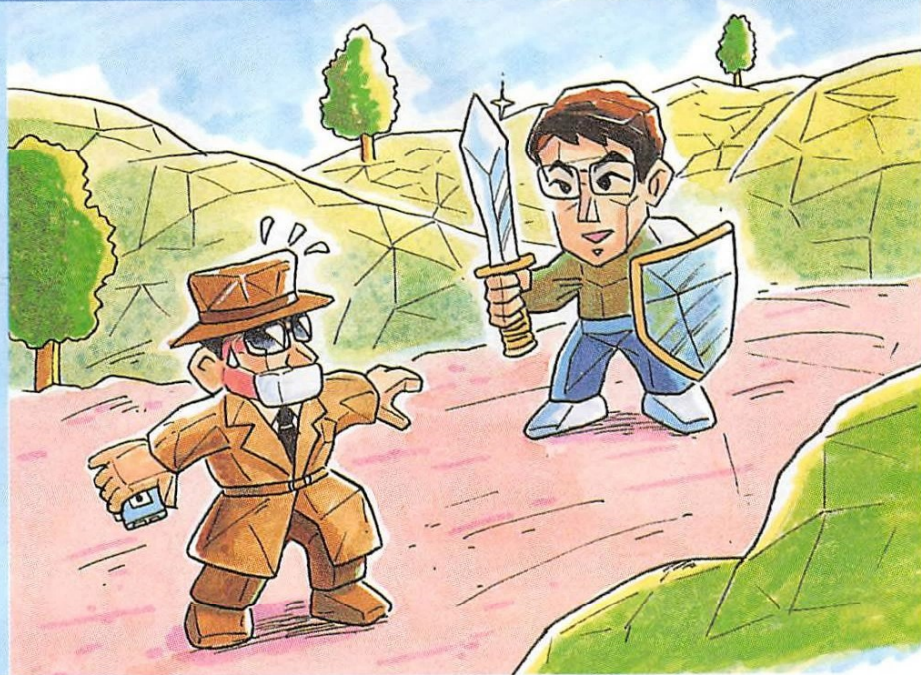
変化したR、G、Bのパレットを画面にセットします（行360）。ここをちょっと工夫するとセピアトーンにすることもできますよ。考えてみましょう。

あわただしい説明になりましたが理解できましたでしょうか？　さて、次回はまだ不明です。ともかく、ラストに向けての2回連続のステップアップ企画を計画中。お楽しみに。

探偵物語（中津さんとホームルーム改め）

私はギャブ＝モブラート。探偵だ。中津の情報を探っている。

さて、新しい情報だが、なにやら開発期間が延びたらしい。この本が出るころにはほぼ終わっているようだが、大幅なバージョンアップをして、より楽しさをアップしたという話だ。

遅くとも3月頃の雑誌には新しい画面写真を載せるつもりであることもキャッチしてる。発売が迫り、あわただしい様子だ。

わかっていることといえば、地形がテクスチャマッピングされたポリゴンを使って描かれている、見下ろし視点のゲームだということである。

倶楽部　～投稿作品より～

【便所の蝶】

小澤孝／東京都・24歳　SCREEN5・4分の1画面サイズ

男にありがちな行動を鋭く突いた（どこがだ！）楽しい作品。人類の自然破壊に警鐘を鳴らしている作品でもあります（大ウソ）。

このコーナーに珍しい、実写取り込み風のタッチもGOOD。単色（ブルー）の背景に、単色（グレー）のキャラクタが、シュールな世界にいざないます。ただプログラムが力技なのが少し残念。

↑うねうね動くタイトルからして、この作品の変態さを象徴している

↑主人公登場！　これは作者だろうか。きっとそうだ、そうに違いない

↑便所にとまっている蝶をみて、思わずにやける。いいキャラクタだよ、ホント

↑ああ。暗い。暗いぞ。でもみんな身に覚えがあるだろう。女性にはないだろうが

↑その蝶がしずくをまき散らしながら、フラフラと接近。ピ～ンチッ！

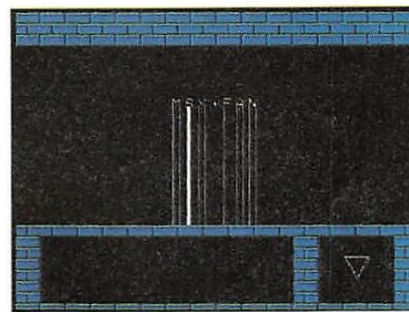
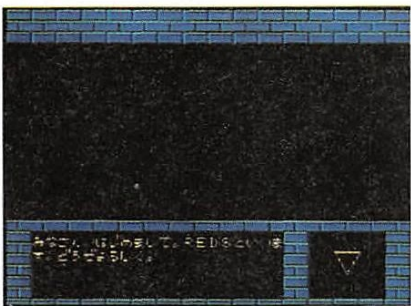
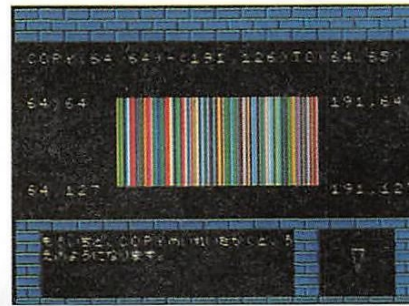
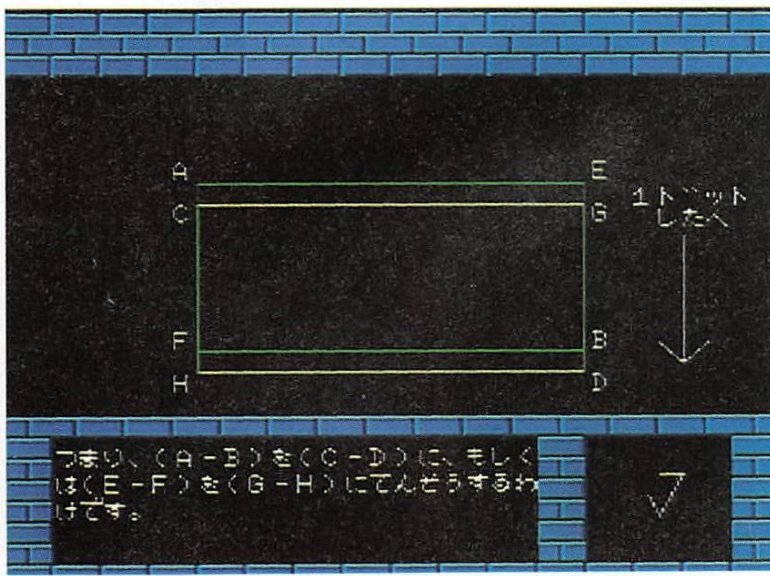
↑今、数々の悪行の報いを受けるときがきた。蝶に代わっておしおきよ

【サークⅡもどきタイトル改】

REDS／山口県・22歳　SCREEN5・全画面サイズ

なんと、プログラムの作り方を教えてくれるCAI（教育）ソフトです。中津のいたらなかったところをフォローしてくれてあって、うれしかったぞ。

中身についてはとりあえず見てください。わかると思います。

これを作るのは大変だったと思いますが、説明もくわしくて非常によろしい。

↑サークⅡのタイトルは下から帯状の模様が次々出てきて、文字になるもの

↑広い窓で数々の説明がなされる。下の窓はメッセージ表示、右下の窓は入力待ち

↑ループを使っていたことが、1つの命令で済んでしまうとは

←ていねいな説明で、サンプルのデモが何度でも見られる工夫もあるし、わかりやすいぞ。応用範囲も実に広くて、ぜひみんなの紙芝居に使ってもらいたいテクニックだ

Mファンにいいたい放題！

整理ができました。それに、「Mファンにいいたい放題！」等のコーナーで、Mファン読者のみなさんの考えていることを知ることができます。思えば、MSXとの出会いはほとんど創刊号の頃。こんなに長いおつきあいの友だちは、本当に数少ない貴重なものです。だから、やっぱり休刊はさびしい。でも、Mファン大好き！　MSX大好き!!　この気持ちはいつまでたっても変わらないと思います。仮に、これが載るとすれば4月号でしょうか。春、いい季節ですね。MSX・FANといっしょに過ごせる春が再び巡ってくることを願っております。（福島県／清野原美橋・17歳）

今月のポイント/3次元から2次元へ

もう今回と次回でこの教室もおしまいです。最後にちょっと難しいテーマに取り組んでみようと選んだのが3Dのアニメーションです。実に奥が深くて、実行する機械もある程度のスペックを必要としますが、MSXで挑戦です。

→P83の「スーパー付録ディスクの使い方」参照

マイクロキャビン
中津泰彦

3次元を計算のレンズで映し出す

どうも、こんにちは。春も終わりですねぇ。と、とぼけたことをいっていますが、例のアレの発売が7月に決定しました！ これを書いている最中はまだマスターアップしていませんが、雑誌には続々と情報が載っているはずです。見てくださいね。

さて今回の紙芝居&動画教室は、予告通り次回へと続く2回連続のステップアップ企画で、ワイヤーフレームを使った3次元のアニメーションにチャレンジします。最近の次世代機では、ポリゴン表示&テクスチャマッピングの3次元格闘ゲームが大流行です。これをMSXで実現するのは速度的に、かなり無理ですが、そのさわりくらいは教えられるかと思います。

よく知っている人は、ワイヤーフレームが、かなり昔に使われていた技術で、時代遅れじゃないかと思うかもしれません。しかし、ばかにすることはできません。見た目はポリゴン&テクスチャマッピングに比べて、少々寂しいですが、それは物体の表示方法が違うだけで、物体の動きとか物体の変形とか、3次元から2次元への座標の変換などについては、何ら変わるところはないのです。その知識は将来3次元の処理をするプログラムを組むときにも、十分使えるので、覚えていて損はないと思います。だいぶ強引かな。

↑→この某次世代機の某ゲームでは、背景の地形がすべてポリゴン&テクスチャマッピングの3D処理。ゲームとしての表現力が向上している……かな

そこでまず今回は基礎編として、シンプルな拡大のみのワイヤーフレームとなります。ですが、それでもプログラムは大きめです。しくしく。

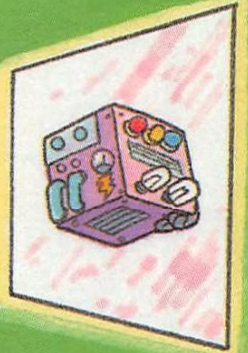
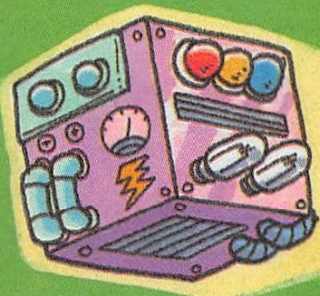
サンプルは、3次元空間上に、MSXの文字が浮かんでいて、それにカメラが近づいていくといったイメージを持ってくだされば、わかると思います。そのMSXの文字は、三角形を基本とする部品を組み合わせて作られています。これは3次元の物体の表現では、よく使われる手法です。3次元空間上の座標は、変換ルーチンを通って、2次元空間（ここではMSXの画面）の座標へと変換されます。

ところで物体はそこにあるだけでなく、どこから見るかが重要です。MSXも前から見るとキーボード、後ろから見るとコネクタ、横から見るとディスクドライブの差し込み口が見えたりして、形が変わりますね。このように、どこから見ているかというカメラ位置の情報は、物体を見るときに非常に重要になってきます。どのようにその情報を処理するのかは、物体の座標からカメラの座標を引き算してやればいいのですが、その説明はちょっと難しくなりますので、そういうものだと思っていてください。

3次元→2次元のかんたんなしくみ

レンズからスクリーンの部分はちょうどカメラのレンズとフィルムとして考えるとわかりやすいでしょう。レンズの部分の計算を変えると、ズームになったり、ゆがんだり、スクリーンの画像も変わるのです。

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の『サーク』でデビュー。以後『FRAY』、『幻影都市』などを手掛けたプログラマ&ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

それでは今回のサンプルプログラムを見ていきましょう。すべての行について見ていくことはせずに、要点だけの説明になるかと思います。

このプログラムをいくつかのまとまったブロックに分けると次のようになります。

- ・準備部（行1000〜行1190）
- ・実行部（行1200〜行1900）
- ・データ部（行2000〜最後）

データ部の情報を元に、準備部で3Dの物体を作り、実行部で実際に動かしています。

ここで重要なのが準備部の前にある、3次元座標から2次元座標に変換する部分です。ここが前のページで述べた、計算のレンズの部分です　（行100〜行990）。

このレンズの部分では、視点を座標軸の原点に持ってくる変換（行110〜行130）のあとに3次元を2次元に表現する変換を行っています。この計算は投影、物体に光を当てて壁にできた影みたいなものですから、そんなにむずかしくはありません。縦横（X、Y座標）の情報はそのままでいいでしょう。ただ奥行き（Z座標）を考慮して、遠くのものは小さくなるように、画面の中心に近くなるようにします（行210、行220）。

この部分はサブルーチンになっていて、実行部からすべての点について呼び出されています。

プログラム解析のために変数の説明をしておきましょう。

PX、PY、PZの配列は元になる3次元座標を覚えます。DX、DYは2次元の座標を覚えます。今回は0〜2しか使ってません。三角形だから。

GPは、グループ分けしたデータの先頭の番号、GNは同一グループの個数を覚えます。

GX、GYは、グループ分けしたデータの表示座標を示します。

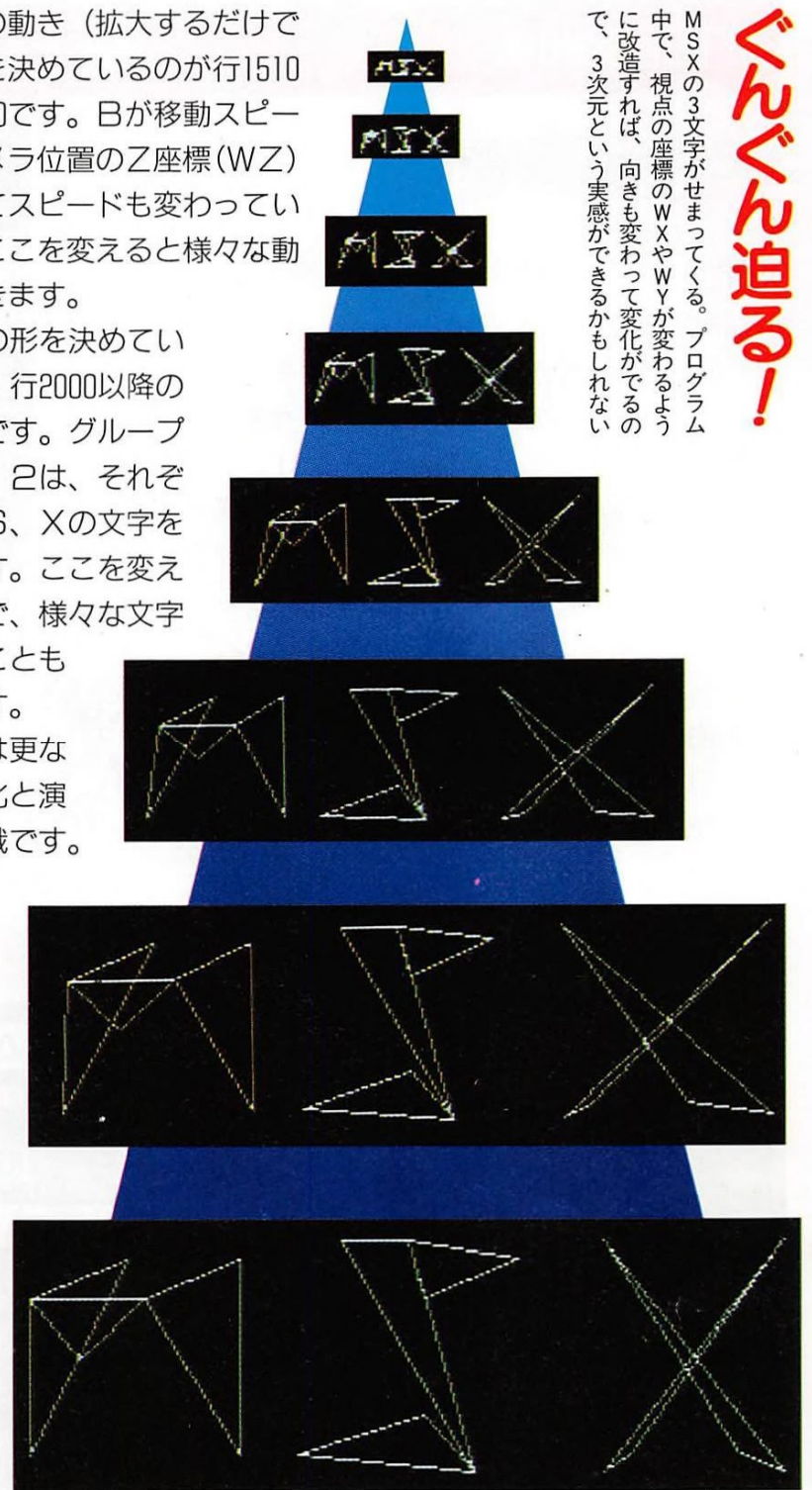
今回は文字が固定なので、グループ分けした意味もあまりないようですが、今後別々に動かすときにはこうした意味が出てくると思います。

行1410が、2次元の3頂点を線で結んで三角形を作っている部分です。ワイヤーフレームがワイヤーフレームたるゆえんの部分です。ここを差し替えて、塗りつぶしたりすると、ポリゴンにもなっちゃいますよ。ただしこのプログラムでは三角形が重なる部分を考慮しないので重なり方が変になるかもしれませんが。

今回の動き（拡大するだけですが）を決めているのが行1510〜行1590です。Bが移動スピードでカメラ位置のZ座標（WZ）によってスピードも変わっています。ここを変えると様々な動きができます。

物体の形を決めているのが、行2000以降のデータです。グループ0、1、2は、それぞれM、S、Xの文字を表します。ここを変えることで、様々な文字を作ることもできます。

次回は更なる高速化と演出に挑戦です。

ぐんぐん迫る！

MSXの3文字がせまってくる。プログラム中で、視点の座標のWXやWYが変わるように改造すれば、向きも変わって変化がでるので、3次元という実感ができるかもしれない

Mファンにいいたい放題！

★NAGI-P　SOFTです。4月号が送られてきたのでお返事します。お久しぶりです。僕は今、東京大学に通っています。今度2年生になります。東大にはすごいコンピュータがありまして、そこでインターネットを通じていろいろ遊んでいるところです。Mファンをみて僕を知っている人もけっこういて、有名人になった気分です。ちなみに今はMacを使っています。Mファンももう終わりなんですね。MSXとは小学校4年生からのつきあいですが、僕にとってはすばらしいコンピュータでした。ずっとがんばってほしいものです。（神奈川県／NAGI-P　SOFT・19歳）

SAMP6. BAS

改造のためのデータ（行2000以降）の説明です。データの1行目は4つの数字からなり、G（物体番号）、NN(物体を構成する三角形の数)、XG、YG(物体の中心XY座標)です。2行目以降は物体各点の、中心からの相対XYZ座標です。3行ずつで三角形を構成し、それがNNセットあります。これで物体1個ぶんです。物体番号ー1で終了です。

```
10 DEFINT A-Z
20 DIM PX(16*3),PY(16*3),PZ(16*3),DX(24)
,DY(24),GP(16),GN(16),GX(16),GY(16)
30 GOTO 1000
100 '2Dへんかん x,y,z wx,wy,wz => xx,yy
110 X1=PX(NM)-WX
120 Y1=PY(NM)-WY
130 Z1=PZ(NM)-WZ
200 ZP=ABS(Z1)
210 DX(J)=X1*64/ZP+128
220 DY(J)=Y1*64/ZP+104
990 RETURN
1000 D=0
1020 READ G,NN,XG,YG
1030 IF G=-1 THEN 1200
1040 GP(G)=D
1050 GN(G)=NN
1060 GX(G)=XG
1070 GY(G)=YG
1080 FOR N9=0 TO NN-1
1090 FOR N=0 TO 2
1100 NM=D*3+N
1110 READ X,Y,Z
1120 PX(NM)=X:PY(NM)=Y:PZ(NM)=Z
1130 NEXT
1140 NN=NN+1
1150 D=D+1
1160 NEXT
1190 GOTO 1020
1200 SCREEN 5
1210 WZ=999:WX=0:WY=0
1220 P=0
1230 SET PAGE P,1-P
1240 P=1-P
1250 CLS
1300 '2dへんかん ヒョウジ゛
1310 FOR G=0 TO 2
1320 WX=GX(G):WY=GY(G)
1330 FOR I=GP(G) TO GP(G)+GN(G)-1
1340 FOR J=0 TO 2
1350 NM=I*3+J
1360 GOSUB 100:'2dへんかん
1370 NEXT
1410 LINE (DX(0),DY(0))-(DX(1),DY(1)),15
:LINE -(DX(2),DY(2)),15:LINE -(DX(0),DY(
0)),15
1420 NEXT
1490 NEXT
1500 'いどう
1510 B=0
1520 IF WZ > 60 THEN B=-10
1530 IF WZ > 100 THEN B=-WZ/5
1540 IF TIME>4 THEN TIME=0
1590 IF B THEN WZ=WZ+B:GOTO 1230
1800 GOTO 1800
1900 END
2000 '3D でーた 3かくけいの3ポイント
2010 '     G, x1, y1, z1
2011 '     G, x1, y1, z1
2012 '        x2, y2, z2
2013 '        x3, y3, z3
2020 DATA 0,  3, 100,  0
2030 DATA   - 0,-40,  0
2040 DATA   -20,-20, 10
2050 DATA   -40,+40,  0
2060 DATA   -20,-20, 10
2070 DATA    20,-20, 10
2080 DATA     0,  0, 10
2090 DATA    40,-40, 0
2100 DATA    20,-20, 10
2110 DATA    40, 40, 0
2120 DATA 1,  3,  0, 0
2130 DATA     0,-40, 10
2140 DATA    40,-40, 0
2150 DATA     5,-20, 10
2160 DATA   -20,-40, 10
2170 DATA     0,-40, 10
2180 DATA    20, 40, 10
2190 DATA     5, 20, 10
2200 DATA    20, 40, 10
2210 DATA   -40, 40, 0
2220 DATA 2,  2,-100,  0
2230 DATA   -40,-40, 10
2240 DATA    40, 40, 10
2250 DATA    20, 40, 0
2260 DATA    40,-40, 10
2270 DATA   -40, 40, 10
2280 DATA   -20, 40, 0
2999 DATA -1,0,0,0
9999 'SAVE"samp6.bas",A
```

Mファンにいいたい放題！

★現実問題として、Mファンの休刊が目前に迫っていて、それは避けられないとは知っていても、悪あがきをしたくなってしまいます。せめてもの抵抗に、1作品でも投稿させていただきます。まだまだ、このまま消えていくつもりはさらさらないです。（長崎県／八木ナナ・19歳）★アンケート結果を見ると、MSXを使っている人は18歳前後がもっとも多いけれど、それにとどまらず、30～50歳の方々もいる。MSXは本当に多くの人たちに愛されているんですね。なんだか、うれしい。でも、MSXはもっと「うれしい」と思っているだろう。（埼玉県／深井直人・20歳）

紙芝居倶楽部 ～投稿作品より～

ミリーの冒険2

REDS・山口県／22歳
SCREEN5・4分の1画面サイズ

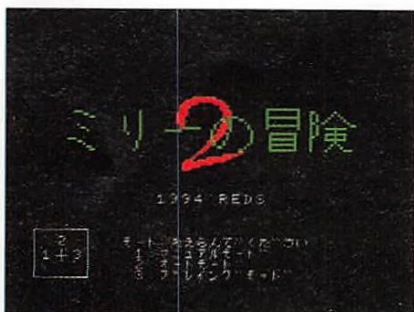
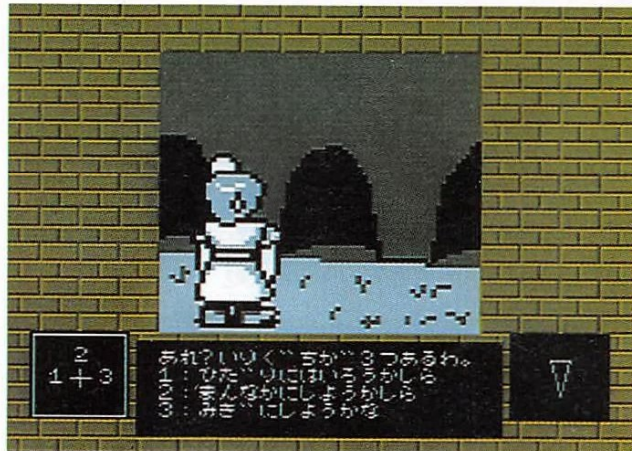
94年8月号で採用された『ミリーのぼうけん』のREDSがアドベンチャーもできる新システムをひっさげての登場だ。前作で好評だった、逆再生や早送りなどのフィーチャーがなくなったのが残念。おまけモードとしてつけられると良かったのだが……。しかし、途中でコマンドを選択したり、ミニゲームがついていたりと、これはすでに1つのゲームといえそうだ。

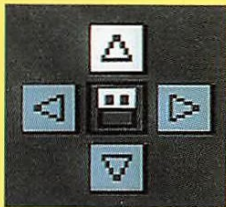
最初にモードを次の3つのうちから選ぼう。

オートモード……キーを押しながら見るだけのモード

マニュアルモード……選択肢を実際に選ぶモード

プレイングモード……さらにミニゲームにも参加するモード

操作はカーソルキーの左、上、右を使用（選択肢の1、2、3にあたります）。それではレッツゴー！

大きいのがメイン、下がメッセージ、右が入力待ち、左がキー選択の各ウインドウ

4つのミニゲーム

・反射神経
光った方向のカーソルキーをすばやく押すゲームだ。間違ったり遅かったりするとどっかーん

・判断力
ひとつひとつのパーツがアニメしてる超難解な16パズル。目を凝らし頭を柔軟にするのだ

・知力
0～9の3つの数字を推測し、その位置を10回までの試行で探し当てないと宝箱は開かない

・体力
崖から落ちそうなミリーをカーソルキーの連打で助けよう。それほど難しくはないです

先生

村上周太・福岡県／21歳
SCREEN5・4分の1画面サイズ

ごーいんなオチの作品で、こーいうのも好きです。駆け足でダァーとネタがやってきて、オチがつくという。こーいうのは、どんどん作ってください。残念ながら付録ディスクには入っていないので、下の写真で見てね。

学校のひとコマ。起立、礼で授業開始

突然一人の生徒が前に出てきて……

先生怒りのチョーク攻撃か？　やっぱり頭を気にしてはいたようだ

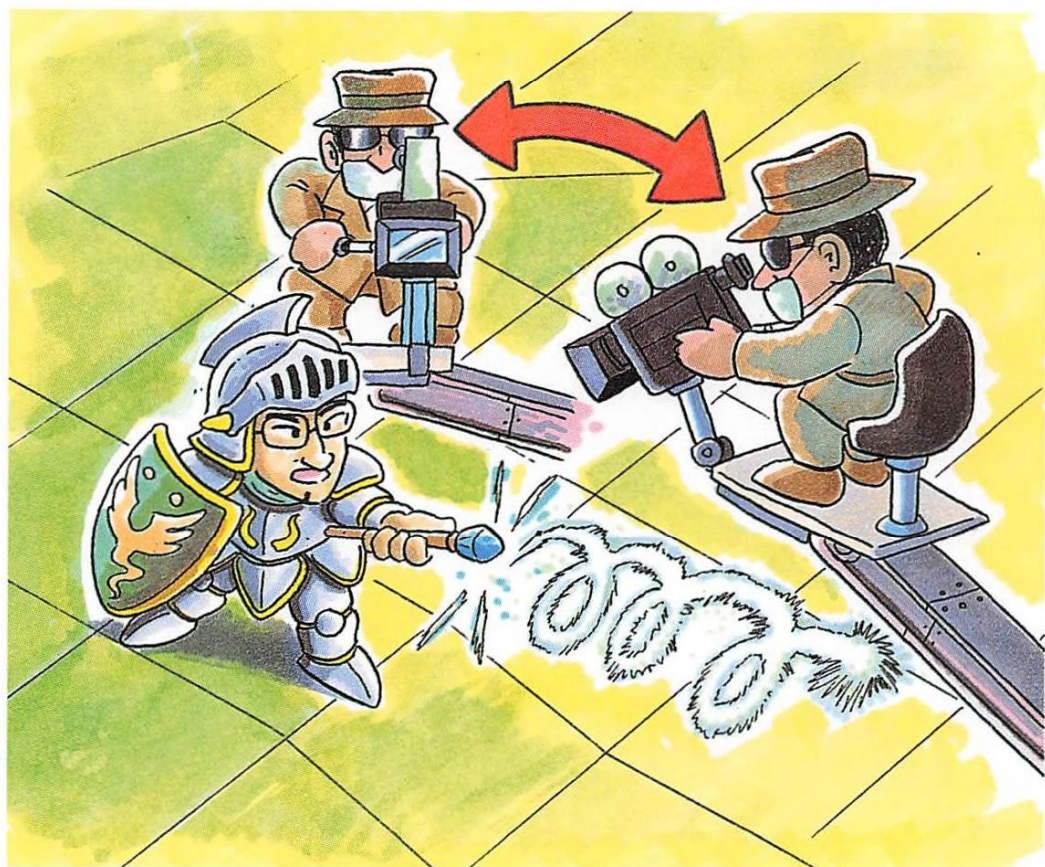
●探偵物語

やあ諸君。私はギャブ＝モブラート、私立探偵だ。中津の行動を探るという依頼を受けている。

ついに長い間謎に包まれていたSRPG（シミュレーションRPGのことのようだ）『リグロードサーガ』のすべてをつかむことができたので、ここに報告したいと思う。

このゲームは、全面的にポリゴンを使ったマップを表示に使用している。そのためいろいろな角度に視点を変えて、360°回転させることもできるということだ。このゲームの魔法エフェクトなどで、カメラワークに活かされて、迫力あるゲームシーンを作っている。

あと、……おっと時間がなくなったようだ。中津はこのゲームのマスターアップに忙しいようだ。次回こそはくわしく説明したいと思う。（編集部注：サターンFANなどの専門誌でチェックしたほうが早いかもしれません）

通常時も戦闘時も画面の切り換えはせず、カメラアングルの変化でアクセントをつける

今月のポイント／動きを研究しよう

3Dのアニメーションでは表示方法も大事ですが、動きとカメラワークが決め手になります。実際の動きをどうプログラムで表現するかがとても難しいのです。これを踏まえ、最終回の今回は3Dアニメーションの神髄に迫る！

→P82の「スーパー付録ディスクの使い方」参照

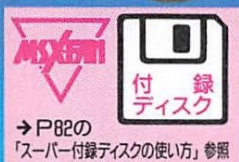
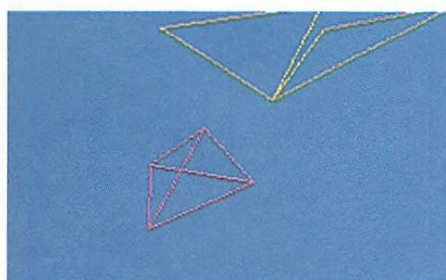
マイクロキャビン
中津泰彦

講師プロフィール

株式会社マイクロキャビン・企画開発課主任。MSX2の「サーク」でデビュー。以後、「FRAY」、「幻影都市」などを手掛けたプログラマ＆ゲームデザイナー。最近は3DOのゲーム開発にも着手して多忙な毎日を送っている。キャビンではMSXにこの人ありといわれるほどMSXとの付き合いが深い。

動きのあるアニメーション

三角形2つでチョウチョの羽なんです

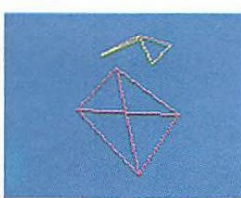
ゆぅ〜っくりと岩に近づいていきます

ふんわりと羽ばたきながら着地します

この教室もラストとなりました。本当に長い間ありがとうございました。この教室でやってきたことが、みなさんの紙芝居&動画人生に役に立ってくれれば、うれしく思います。

さて今回は、この教室最大にして最強のプログラムとなりました、3Dの第2回をお送りします。

ちなみに、某ゲームはマスターアップしました／　くわしくはサターンFANなどの雑誌を見てくださいね。この3Dゲームの技術は、今回のプログラムと同じ技術上に存在しますので、この方向を目指す人にも役立つと思います。

前回は、3Dらしい回転が入っていませんでしたが、今回はグルングルン回してみましたし、動きのあるアニメーションもやっています。

動き。みなさんも、3Dアニメに取り組むなら、動きについて色々と工夫するとおもしろいですよ。物の動き一つとっても、かんたんにはできなくて、奥の深い世界だと感じるはずです。ビデオで動きをチェックしたり、物理の本で勉強するのも一つの手ですね。

今回も前回と同じく三角形を基本としています。前回と比べると、処理のとちゅうにSINやCOSの計算が入って、やたらと遅くなりますので、プログラムのスタイルを多少変更して、三角形の数は思いっきり減らしました。が、シンプルななかに美しさをねらったサンプルプログラムです。

プログラムは、3Dの計算と表示を分離しました。MSX2／2+の人は計算にだいぶ時間がかかりますので、ティーブレイクでもしてお待ちください。

ちなみに、ワイヤーフレームの計算でクリッピングと呼ばれる処理をしていませんので、プログラムを改造すると計算エラーすることがあります。一応覚えておいてください。もちろん、改造はどしどしやってみてくださいね。

このプログラムもアセンブラ化すればスピーディになりますから、リアルタイムの計算、表示にもたえるかもしれません。

この数字が0になるまで待つぞ。0まで待つぞ。0まで待つぞ

クリッピングとは

今回の例でいうと、カメラが蝶や岩に近づきすぎると頂点が画面からはみ出ます。なにも対策を施さないとエラーが出たり、直線が変な位置になったりします。そこで、画面の外に頂点があるかのように、点を決めて線を引く位置を調節することをクリッピングといいます。次世代機だと数千ドット分くらいは自動でしてくれて楽なんですが。

動きを数式に変換する

物体の形や動きを、より少ないデータや関数で表現するセンスが必要だ。今回の蝶の羽ばたく動きは三角関数を利用している。動きをそのままサンプリングするモーションキャプチャも最近の流行ではあるが。

今回のテーマは、蝶と岩のつもりです。プログラムが大きいのでザザッと説明しますので、ついてきてくださいね。

今回のプログラムは次のブロックに分かれます。

- 初期化(行10〜90、1000〜1170)
- 3Dサブルーチン(行100〜580)
- 3D計算処理(行1210〜1470)
- 2D表示処理(行1800〜1999)
- 3D計算サブルーチン(行3000〜4090)
- 3Dデータ(行5000〜5460)
- 2D表示サブルーチン(行6000〜6190)

3Dのデータを2Dの画面に変換する処理は、前回とまったく同じです。そこで今回は前回と異なった部分、回転と動きにスポットを当ててみましょう。

その前に大まかな流れの説明です。まず最初に下準備として、蝶や岩の各点の座標の読み込みや、計算に使う三角関数テーブルの作成です。

岩に止まった蝶。羽ばたきも心持ち小さくなっているような気がしませんか

拡→大

カメラも蝶にグングン近づいていきます。ちょっとご機嫌ななめかな

回→転

するといきなり視点が岩の周りをグルグル回転し始めます。これぞ3Dアニメ

次にメインの3D処理です。静止画1枚ごとに新しい座標と角度を使って、それぞれの点の座標を割り出し、DX、DYの配列にドンドン溜めます。メインループ(行1300〜1400)は蝶が岩に止まって(蝶と岩の距離が0になるかどうかで判定。行1350)、上下に一回転するまで(回転角度C2で判定。行1400)、メインループ2は蝶が飛び立つ部分を処理しています。

最後に溜めておいたデータを元に実際に表示をして終わりです(行1800〜1999)。

さて、回転です。

このサンプルではカメラの角度を変えるというよりも、カメラの角度は固定して、世界そのもののほうを回しています。結局は同じことなのですが、このほうが対象を画面に捕らえるためには楽だからです。実際には蝶や岩の頂点一つずつについてZ軸、X軸を中心に回転変換をしています(行300〜580)。

動きの説明です。

まずカメラの動きからです。

C3は常に変化していて、カメラが岩のまわりを旋回してるようすを表しています。C2は蝶と岩の距離が0になった時点で変化を始め、上下の回転を表しています。CZはカメラのZ座標です。これも蝶と岩の距離が0になってから変化します。最初はその差分TZが−1なので近づいていき、いったん距離90まで近づいた後はTZを1にして離れていきます。蝶が飛び立ってからはカメラは固定です。

次に蝶の動きです。

蝶の位置は岩との距離によって決まっていて、止まる場所からX方向:Z方向=2:1の比率で飛んでいます。蝶の羽ばたきは、三角関数を使用しています。これによって、上端下端ではゆっくり、中心に近いところでは速くという動きになります。

以上で解説を終わります。本当にありがとうございました。みなさんにはゲームでまたお会いしましょう！

変数リスト

■主な変数

DX(P)、DY(P)　計算した2Dデータの保存。Pは点の通し番号

PX(D)、PY(D)、PZ(D)　各頂点の座標。Dは0〜8で頂点番号

LN(N, M)　線を構成する頂点の番号。Nは線の番号。Mは0か1。Mが0と1(始点、終点)の2つセットで一つの線になる

DP　頂点番号のオフセット(画面1枚ごとに9増える)

K1、K2　蝶と岩の色

■蝶の変数

BX、BY、BZ　蝶の座標

BL　蝶と岩の距離

BO　BLの差分

BH　羽ばたき用角度

■カメラの変数

CX、CY、CZ　カメラ座標

C1、C2、C3　カメラの角度

CP　回転角度の差分

TZ　カメラ座標差分

■計算用変数

SN(R)、CS(R)　三角関数テーブル。小数にならないようにあらかじめ100倍してある。Rは0〜71で、5倍すると角度になる

R、RR　回転処理用の角度

そしてさようなら〜

ゆっくりと岩を離れる蝶。羽の上下に合わせた微妙な上下動が、ワイヤーフレームの蝶に生命を吹きこんでいるのです。というわけでさようなら

SAMP8. BAS

```
10 DEFINT A-Z
20 DIM PX(10),PY(10),PZ(10)
30 DIM LN(16,2)
40 DIM SN(72),CS(72)
50 DIM DX(2000),DY(2000)
60 SCREEN 1
90 GOTO 1000
100 '2dへんかん x,y,z  => xx,yy
110 X1=X-CX
120 Y1=Y-CY
130 Z1=Z-CZ
140 ZP=ABS(Z1)
150 XX=X1*100/ZP+128
160 YY=Y1*100/ZP+104
170 RETURN
200 'かいてんへんかん
201 ' a,b,c, r => aa,bb,cc
202 ' cじ゛くをちゅうしんに かいてん
210 RR=(R/5+720) MOD 72:'ざ゛じょうほせい
220 AA=(A*CS(RR)-B*SN(RR))/100
230 BB=(A*SN(RR)+B*CS(RR))/100
240 CC=C
250 RETURN
300 '3d-2dへんかん ちょうちょ
310 X=X+BX:Y=Y+BY:Z=Z+BZ
500 '3d-2dへんかん さんかくすい
510 A=X:B=Y:C=Z:R=C3
520 GOSUB 200
530 A=CC:B=BB:C=AA:R=C2
540 GOSUB 200
560 X=CC:Y=BB:Z=AA
570 GOSUB 100
580 RETURN
1000 FOR D=0 TO 8
1010  READ PX(D),PY(D),PZ(D)
1020 NEXT
1030 FOR I=0 TO 11
1040  READ LN(I,0),LN(I,1)
1050 NEXT
1100 'sinテーブ゛ルさくせい
1110 FOR R=0 TO 355/5
1120  READ SN(R)
1130 NEXT
1140 FOR I=0 TO 355/5
1150 R=(90/5+I) MOD 72
1160 CS(I) = SN(R)
1170 NEXT
1210 PRINT "3Dのけいさんをします. おまちくだ゛さい."
1220 CX=0:CY=0:CZ=150
1230 C1=1:C2=310:C3=0
1240 BL=140:BR=0:BH=0
1250 GOSUB 3300:'ショキカ
1260 CP=-1:TZ=-1:BO=-3
1290 P=0:DP=0
1300 'メインループ゜
1310  GOSUB 4000:'さんかくすい
1320  GOSUB 3000:'てふてふ
1330  DP=DP+9
1340  PRINT INT((1737-DP)/9),
1350  IF BL<>0 THEN 1390
1360  C2=C2+CP
1370  IF C2<290 THEN CP=5
1380  CZ=CZ+TZ
1385  IF CZ<90 THEN TZ=1
1390  C3=C3+10
1400 IF C2<(320+360) THEN1300
1410 BO=3
1420 'メインループ゜2
1430  GOSUB 4000:'さんかくすい
1440  GOSUB 3000:'てふてふ
1450  DP=DP+9
1460  PRINT INT((1737-DP)/9),
1470 IF BL<140 THEN 1420
1800 SCREEN 5:COLOR 15,5
1805 K1=10:K2=9
1810 FOR D=0 TO DP-9 STEP 9
1820  SET PAGE P,1-P
1830  P=1-P
1840  CLS
1850  GOSUB 6000:'ひょうじ゛
1860 NEXT
1870 GOTO 1800
1999 END
3000 'ちょうちょ へんかん
3010 FOR I=4 TO 8
3020 X=PX(I):Y=PY(I):Z=PZ(I)
3030 GOSUB 300
3040 DX(I+DP)=XX:DY(I+DP)=YY
3050 NEXT
3200 'はね を うご゛かす
3210 Z=20*SN(BH)/100
3220 IF BL<>0 THEN BH=BH+4 ELSE BH=BH+2
3230 BH=BH MOD 72
3240 PZ(4)=Z:PZ(5)=Z:PZ(7)=Z:PZ(8)=Z
3300 'いど゛うする
3310 BR=(BR+1) MOD 72
3320 BL=BL+BO:IF BL < 0 THEN BL=0
3330 BX=BL/1.5
3340 BY=0
3350 BZ=BL/3+50
3490 RETURN
4000 'さんかくすい へんかん
4010 FOR I=0 TO 3
4020 X=PX(I):Y=PY(I):Z=PZ(I)
4030 GOSUB 500
4040 DX(I+DP)=XX:DY(I+DP)=YY
4050 NEXT
4090 RETURN
5000 'ザ゛ヒョウ
5005 '  さんかくすい
5010 DATA 0,-40,0
5020 DATA -40,40,0
5030 DATA 40,40,0
5040 DATA 0,0,50
5045 '  ちょうちょ
5050 DATA -20,-20,0
5060 DATA 30,-15,0
5070 DATA 0,0,0
5080 DATA -20,20,0
5090 DATA 30,15,0
5100 'ラインデ゛ータ
5105 '  さんかくすい
5110 DATA 0,1
5120 DATA 1,2
5130 DATA 2,0
5140 DATA 0,3
5150 DATA 1,3
5160 DATA 2,3
5165 '  ちょうちょ
5170 DATA 4,5
5180 DATA 5,6
5190 DATA 6,4
5200 DATA 7,8
5210 DATA 8,6
5220 DATA 6,7
5300 'sinデ゛ータ 5ド゛ きざ゛み
5310 DATA 00,08,17,26,33,42
5320 DATA 50,56,64,70,76,81
5330 DATA 86,90,94,96,98,99
5340 DATA 100,99,98,96,94,90
5350 DATA 86,81,76,70,64,56
5360 DATA 50,42,33,26,17,08
5370 DATA 000,-08,-17,-27
5380 DATA -33,-42,-50,-56
5390 DATA -64,-70,-76,-81
5400 DATA -86,-90,-94,-96
5410 DATA -98,-99,-100,-99
5420 DATA -98,-96,-94,-90
5430 DATA -86,-81,-76,-70
5440 DATA -64,-56
5450 DATA -50,-42,-33,-26
5460 DATA -17,-08
6000 'ちょうちょ  ひょうじ゛
6010 FOR I=6 TO 11
6020 L1=LN(I,0)+D:L2=LN(I,1)+D
6030  LINE (DX(L1),DY(L1))-(DX(L2),DY(L2
)),K1
6040 NEXT
6100 'さんかくすい ひょうじ゛
6110 FOR I=0 TO 5
6120 L1=LN(I,0)+D:L2=LN(I,1)+D
6130  LINE (DX(L1),DY(L1))-(DX(L2),DY(L2
)),K2
6140 NEXT
6190 RETURN
9999 'SAVE"samp8.bas
```

Mファンにいいたい放題！

★Mファンよ、さようなら、そしてありがとう！　これからのMSXの歴史は、俺たちユーザーが作っていきます。本当に、本当に、ごくろうさまでした&ありがとうございました。とりあえず、みなさん(特にOrcさん(笑))夏コミで会いましょう!!(埼玉県／宮崎幸雄・19歳)★このまえ中古屋にいったら、MSX関連のものが値上がりしていた。まだまだ需要があるということなんだろう。(京都府／夢見草・18歳)★今思うと、3年前はじめてMファンを見つけ、ねだって買ってもらったあの日がよみがえってきます。今までありがとう、そしてさようなら、Mファン！(岩手県／NV組HAMU・12歳)

AFTER SCHOOL

紙芝居倶楽部

～投稿作品より～

VANITY ★

時は白亜紀。恐竜たちが地球を支配していた時代の話……。

シダの葉陰からイグアノドンの群れを狙う目が光った。茂みをかき分けて3頭のヴェロキラプトルが飛び出してきた。イグアノドンの群れは一斉に逃げ出すが、足の遅い一頭がターゲットに……。命を懸けたバトルの末に恐竜たちが見た物は!?

作者の手紙より（ほぼ原文ママ）。

HATABO！／神奈川県／22歳
SCREEN5／16分の1画面サイズ

中津「恐竜が走り回るパワフルな作品。技術的にもいろいろな画面合成を使っていて、興味深く見させてもらった。動画のテクニックも及第点だ！　これからも精進してほしい」

うさぎパワフル ★

童話「うさぎとかめ」のパロディ。うさぎが亀に再戦を申し込むが……。パレット切り換えを利用し、VRAM128KのMSXで、SCREEN5フル画面で12枚ぶんを、ディスクを読まずに表示することができる。

中津「楽しい作品だ！　こういうノリは大好きだぞ。パレット切り換えのテクニックも使っているんだ。締め切りに間に合ってよかったね」

沢田外道／岩手県／23歳
SCREEN5／全画面サイズ

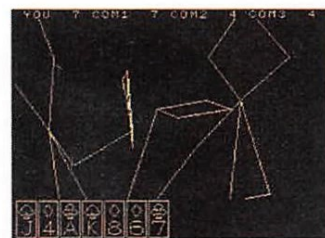
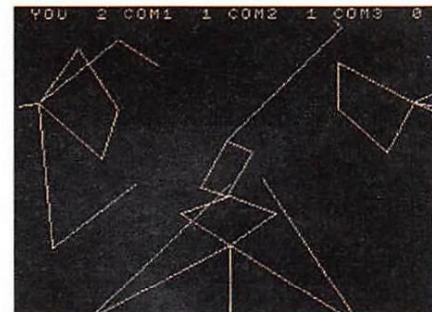
←COLOR＝NEWするとこんなかんじ

↑うさぎって……。月野うさぎよりいいか

バーチャルババヌキ ★

ババヌキのようすを3Dで表現。1～4キーで視点切り換えで、それぞれプレイヤー、背後霊、電球、見上げ視点。ESCキーでカードの表示、非表示。リプレイはスペースキー。

中津「アセンブラは速くていいな。あおざかな君は、紙芝居教室にない作品をということだが、残念だが同じ方向の作品になったみたいだ。読みが甘かったな。でも採用だ。おめでとう」

あおざかな／福岡県／20歳
SCREEN5／全画面サイズ

←やはり動きと視点に苦労したそうだ

↑見上げ視点。下からのアングルが迫力

★探偵物語改め探偵日記

さて、これが最後の報告となる。私は、私立探偵のギャブ＝モブラートだ。

中津が開発にかかわっていたサターンのRPG、リグロードサーガについて最終的にわかったことを報告しよう。

発売は、7月21日だそうである。内容については、専門の雑誌のほう（編集部注：サターンFANなどをご覧ください）にくわしい情報が掲載されているようなので省略するが、一つおもしろい情報がはいった。

サターンで苦労したのは、やはりメモリだそうである。CD-ROMだからメモリがたくさんあっていいぞ～と、ドカドカ入れたら、すぐいっぱいになってしまい、ヒイヒイと苦労してデータを減らしたらしい。次世代機になってもメモリが楽にならない、なぜだろう？　……と周囲に洩らしていたようだ。

さあ、これですべて終了した。今回の任務は今までの仕事のなかでも、興味深いものだった。また私が必要になったらぜひ連絡をくれたまえ。じゃあな。

……とギャブ＝モブラートは闇のなかへと消えていくのであった。　～Fin～

Mファンにいいたい放題！

★私のファンダム投稿作品はボツのようですね。まあ、しかたないと思うけどな～。『もっとあおざかな』はMファンに載るために作ったので（傾向と対策）、「載って当然」と思っていたのですが、ボツになったら死んでたかもしれません。あとは『バーチャルババヌキ』に賭けるしかないな。競走率2倍？（福岡県／青魚・20歳）★いつかMSX3がでたら、MSX・FAN復活しますよね。今までどおりの編集者たちで。MSX3出ますよ、たぶん。いやきっと。いやいや絶対出ます！　では、その日に会いましょう。（宮城県／鴇田久良・22歳）